

accenture

C / C S

(Customer Information Control System)





Contenido

Introducción a CICS	<ul style="list-style-type: none"> Conceptos Básicos Funciones que realiza
Comunicación con programas on-line	<ul style="list-style-type: none"> Ciclos Areas de Comunicación Link XCTL
Interfase de Programación de Alto Nivel (HLPI)	<ul style="list-style-type: none"> Tareas Condiciones de excepción
Mapas	<ul style="list-style-type: none"> Físico Simbólico Macros para su generación Atributos
Mandatos para el Control de Terminales	
Manejo de archivos VSAM	
Browsing y Colas TS	
Arquitectura Altamira	



Contenido

Introducción a CICS

Comunicación con programas On-Line

Interfase de Programación de Alto Nivel (HLPI)

Mapas

Mandatos para el Control de Terminales

Manejo de archivos VSAM

Browsing y Colas TS

Arquitectura Altamira

Introducción

Conceptos Básicos

Funciones específicas



CICS (Customer Information Control System)

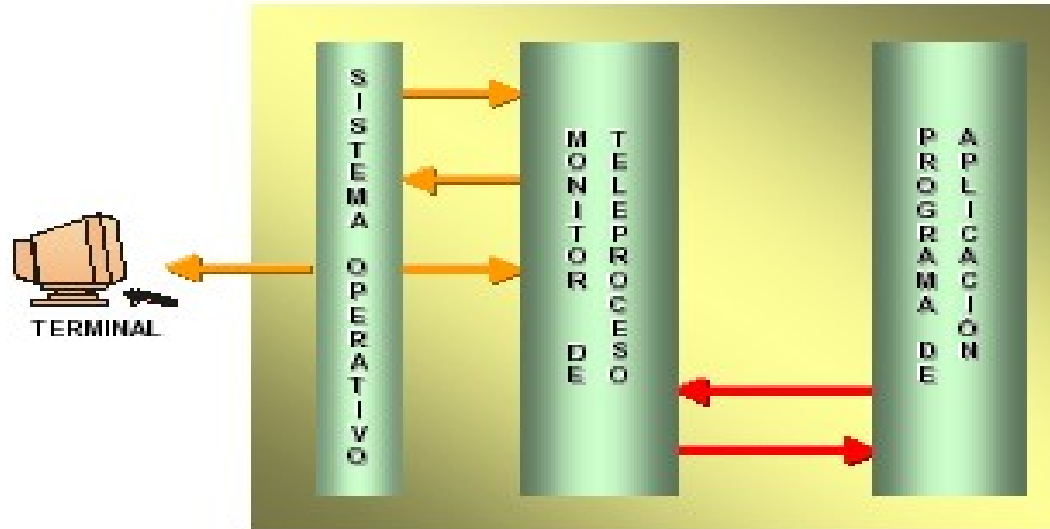
Monitor transaccional que gestiona el paso de una transacción entre etapas subsecuentes y asegura su ejecución completa.

Permite a los programadores :

- ✓ Desarrollar pantallas genéricas sin importar el terminal utilizado.
- ✓ Definir las entidades on-line necesarias para las aplicaciones (transacciones, programas, mapas).
- ✓ Realiza el balance de cargas de trabajo al distribuir el procesamiento a lo ancho de una red de computadoras, evitando la sobrecarga de un dispositivo específico.
- ✓ Permite la distribución del acceso a las aplicaciones.



Comunicación entre un programa, el CICS y el Sistema Operativo





Contenido

Introducción a CICS

Comunicación con programas On-Line

Interfase de Programación de Alto Nivel (HLPI)

Mapas

Mandatos para el Control de Terminales

Manejo de archivos VSAM

Browsing y Colas TS

Arquitectura Altamira

Introducción

Conceptos Básicos

Funciones específicas



Conceptos Básicos

Cics trabaja bajo una serie de conceptos básicos los cuales son los siguientes :

Transacción

Tarea

Multitarea

Multiprogramación

Reentrancia

Es la manera de identificar una función que se pide que haga el Sistema Operativo, por ejemplo (dar de alta un contrato, consultar el saldo de una cuenta corriente, etc.).

Una transacción lleva asociada informaciones tales como :

- Prioridad
- Plan de acceso a un gestor de base de datos
- Ejecución de un programa.

Una transacción se identifica en el sistema con un código alfanumérico, que en el caso del CICS es de una cadena de cuatro caracteres.



Conceptos Básicos

Cics trabaja bajo una serie de conceptos básicos los cuales son los siguientes :

Transacción

Tarea

Multitarea

Multiprogramación

Reentrancia

¿Qué ocurre cuando varios usuarios solicitan ejecutar la misma transacción?

¿Cómo podría CICS manejar varias peticiones a la vez?

CICS puede manejar las múltiples peticiones asignando a cada usuario una tarea, la cual, es una unidad de trabajo a completar. Al procesar varias peticiones, se crea una tarea para cada una de ellas.

Una tarea siempre estará en uno de los dos estados siguientes :

- Activa (en ejecución)
- Suspendida (está esperando algo del sistema).



Conceptos Básicos

Cics trabaja bajo una serie de conceptos básicos los cuales son los siguientes :

Transacción

Tarea

Multitarea

Multiprogramación

Reentrancia

Es una de las características del CICS. Significa que se permite que varias tareas se ejecuten simultáneamente con técnicas de tiempo compartido.

Cuando existe un único procesador sólo una tarea puede estar ejecutándose realmente en cada momento, aunque al usuario le parecerá que se ejecutan varias a la vez. Esto se consigue dividiendo el tiempo de la CPU entre una y otra tarea, y repartiendo el tiempo de proceso entre varios programas.

Es lo que se llama **Multiprogramación** con tiempo. La auténtica multitarea se consigue solamente con procesadores en paralelo (multiproceso).



Conceptos Básicos

Cics trabaja bajo una serie de conceptos básicos los cuales son los siguientes :

Transacción

Tarea

Multitarea

Multiprogramación

Reentrancia

Un programa debe codificarse de forma que pueda ser compartido por más de una transacción a la vez, en este caso, si hay más de una tarea que está compartiendo el programa, el código deberá permanecer siempre inalterable.

Este tipo de codificación, que permite que más de una tarea comparta un programa, y que no se puede modificar, se conoce como codificación **reentrante**.

CICS permite compartir la Procedure Division de un programa y que cada tarea acceda a *Working-Storage's* diferentes, con las mismas definiciones de datos, pero con diferentes contenidos en sus variables.



Contenido

Introducción a CICS

Comunicación con programas On-Line

Interfase de Programación de Alto Nivel (HLPI)

Mapas

Mandatos para el Control de Terminales

Manejo de archivos VSAM

Browsing y Colas TS

Arquitectura Altamira

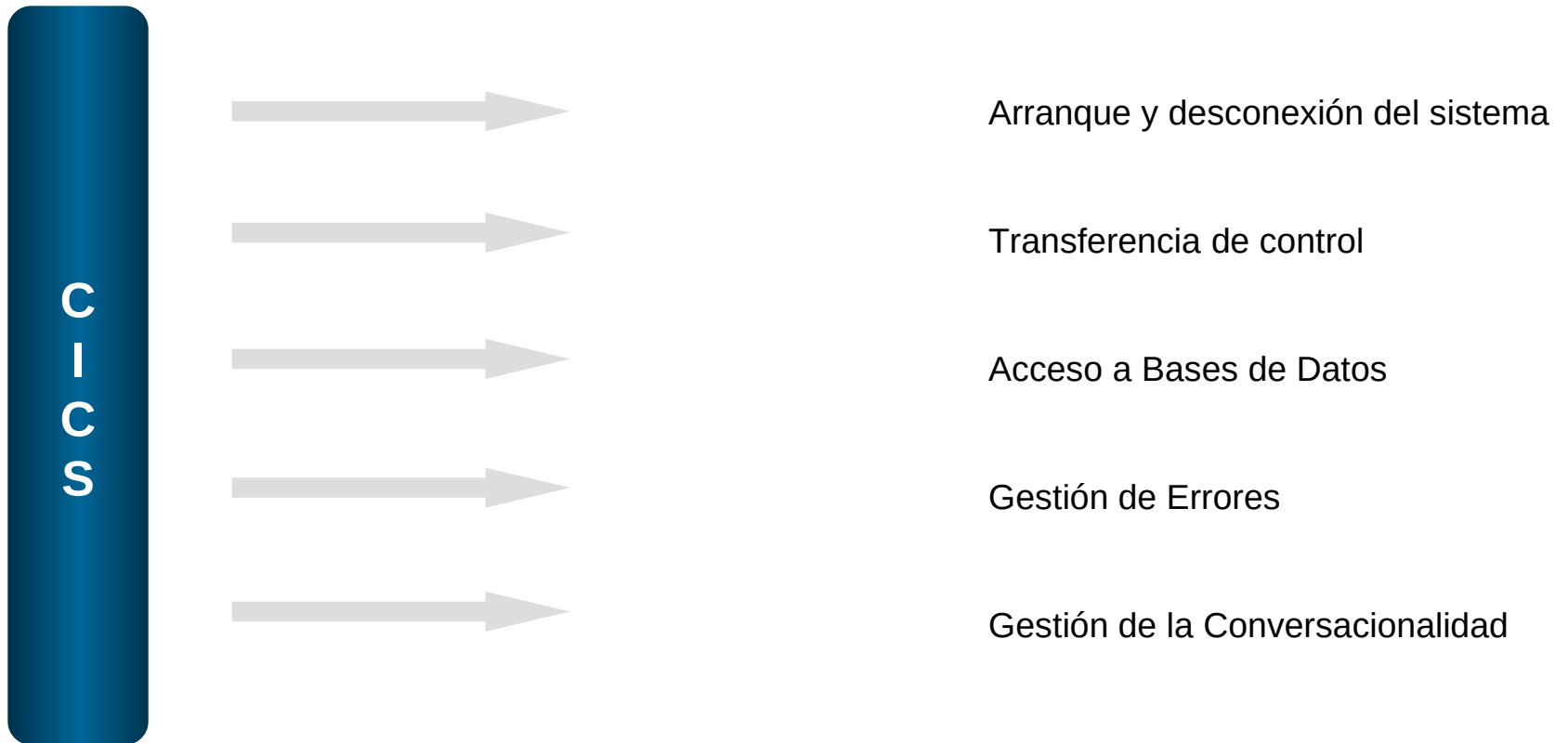
Introducción

Conceptos Básicos

Funciones específicas



Funciones que realiza CICS





Funciones que realiza CICS

Arranque y desconexión del sistema

Transferencia de control

Acceso a Bases de Datos

Gestión de Errores

Gestión de la Conversacionalidad

Esta función vincula las tareas a realizar al comienzo y final de la sesión del día.

Estas tareas son :

- ✓ Abrir y cerrar archivos
- ✓ Activar y desactivar las líneas de comunicación

Es desde CICS donde se abren y cierran los archivos y no desde el código del programa de aplicación.



Funciones que realiza CICS

Arranque y desconexión del sistema

Transferencia de control

Acceso a Bases de Datos

Gestión de Errores

Gestión de la Conversacionalidad

CICS gestiona la transferencia del control entre diferentes programas de aplicación en el entorno online.

Por ejemplo :

- Transferir el control a una rutina.

En un programa online multimodular, las sentencias CALL que transfieren el control son sustituidas automáticamente por comandos CICS.



Funciones que realiza CICS

Arranque y desconexión del sistema

Transferencia de control

Acceso a Bases de Datos

Gestión de Errores

Gestión de la Conversacionalidad

En CICS se pueden manipular bases de datos, accediendo a las mismas, ya sea para realizar consultas directas, barrido de las bases de datos, actualizar su contenido, insertar nuevos registros o borrarlos físicamente.



Funciones que realiza CICS

Arranque y desconexión del sistema

Transferencia de control

Acceso a Bases de Datos

Gestión de Errores

Gestión de la Conversacionalidad

CICS contiene facilidades para capturar el error producido, terminar la transacción y permitir que las demás transacciones continúen su proceso.

Esto permite al usuario introducir la transacción de nuevo, sin que afecte a otros usuarios del sistema.

No es práctico que el programa maneje todos los errores condicionales que pueden surgir en un entorno online.



Funciones que realiza CICS

Arranque y desconexión del sistema

Transferencia de control

Acceso a Bases de Datos

Gestión de Errores

Gestión de la Conversacionalidad

CICS puede permitir a muchos usuarios conectarse y usar el sistema online concurrentemente, pero el sistema tiene una memoria de trabajo limitada.

Hay dos tipos de técnicas de programación :

Conversacional

Pseudoconversacional

que afecta cómo el monitor utiliza la memoria principal del sistema.



Funciones que realiza CICS

Arranque y desconexión del sistema

Transferencia de control

Acceso a Bases de Datos

Gestión de Errores

Gestión de la Conversacionalidad

Programación conversacional

Técnica que resulta cuando un programa es cargado en memoria y permanece allí hasta que el usuario finaliza el trabajo con el programa.

El programa no es descargado de la memoria cuando no está activo.

Los programas conversacionales pueden ser utilizados en un bajo volumen de situaciones cuando una gestión eficiente de la memoria no es proporcionada.

Funciones que realiza CICS

Arranque y desconexión del sistema

Transferencia de control

Acceso a Bases de Datos

Gestión de Errores

Gestión de la Conversacionalidad

Programación Pseudoconversacional

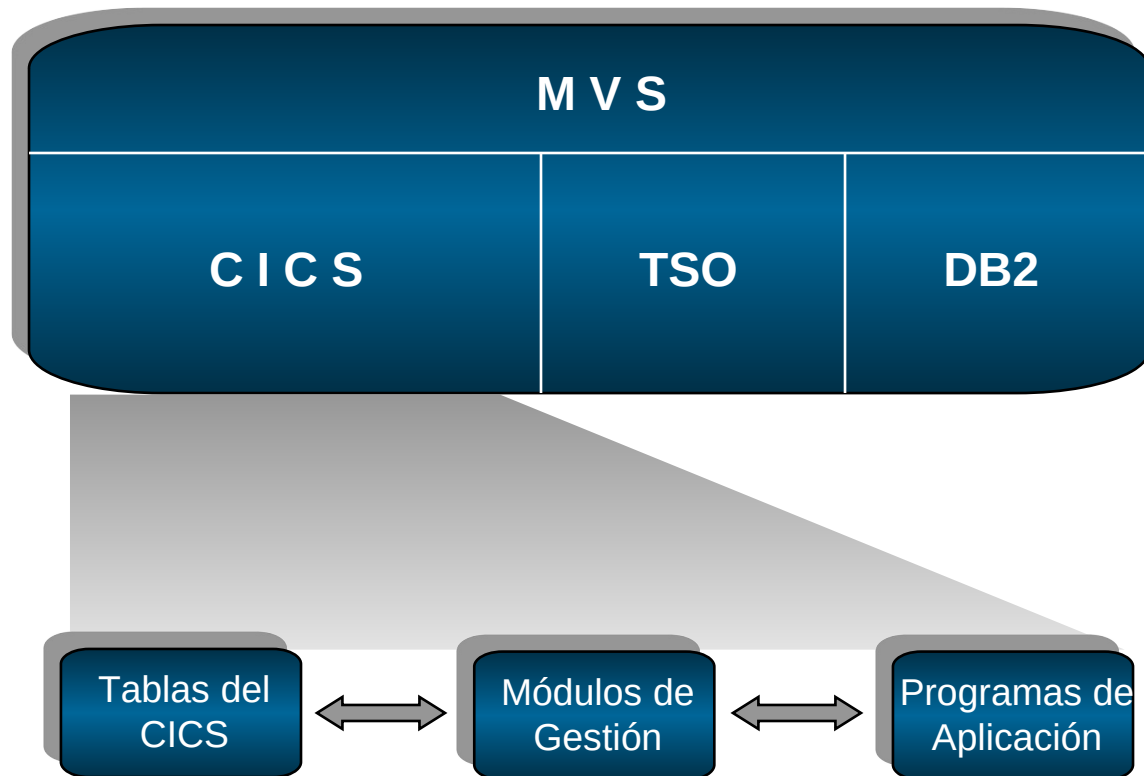
El programa no gestiona el envío y recepción de manera continua, sino que realiza desconexiones cada vez que necesita una respuesta por parte del usuario, suspendiendo temporalmente la ejecución de la tarea y liberando los recursos utilizados por la misma.

Este modo requiere una manera especial de codificación de los programas de aplicación.

Es necesario establecer una serie de ciclos que permiten a los programas proseguir las tareas a partir del punto en que se realizó la última desconexión.



Principales componentes del CICS





Contenido

Introducción a CICS

Comunicación con programas On-Line

Interfase de Programación de Alto Nivel (HLPI)

Mapas

Mandatos para el Control de Terminales

Manejo de archivos VSAM

Browsing y Colas TS

Arquitectura Altamira

Ciclos

Areas de Comunicación

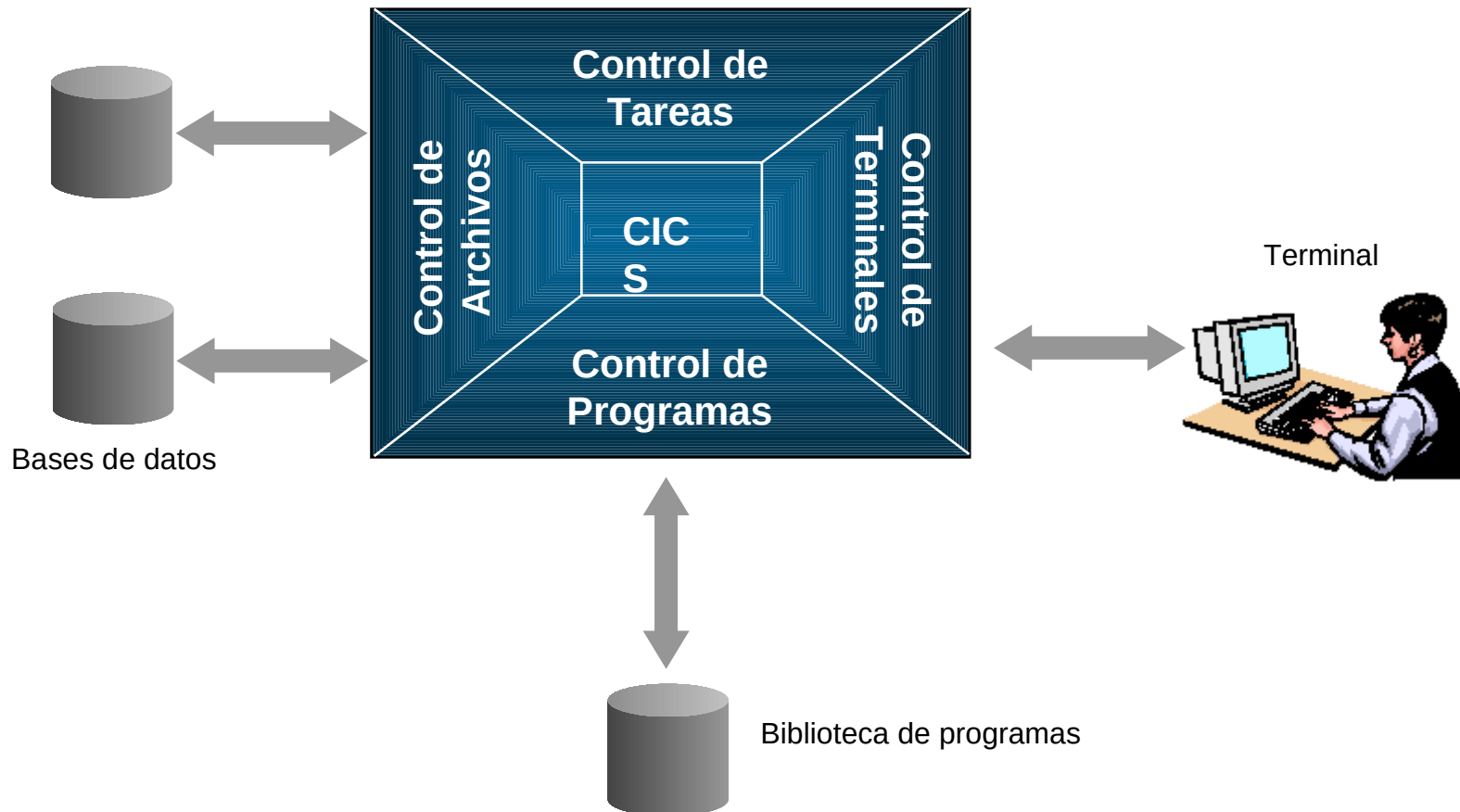
Link

XCTL

Return



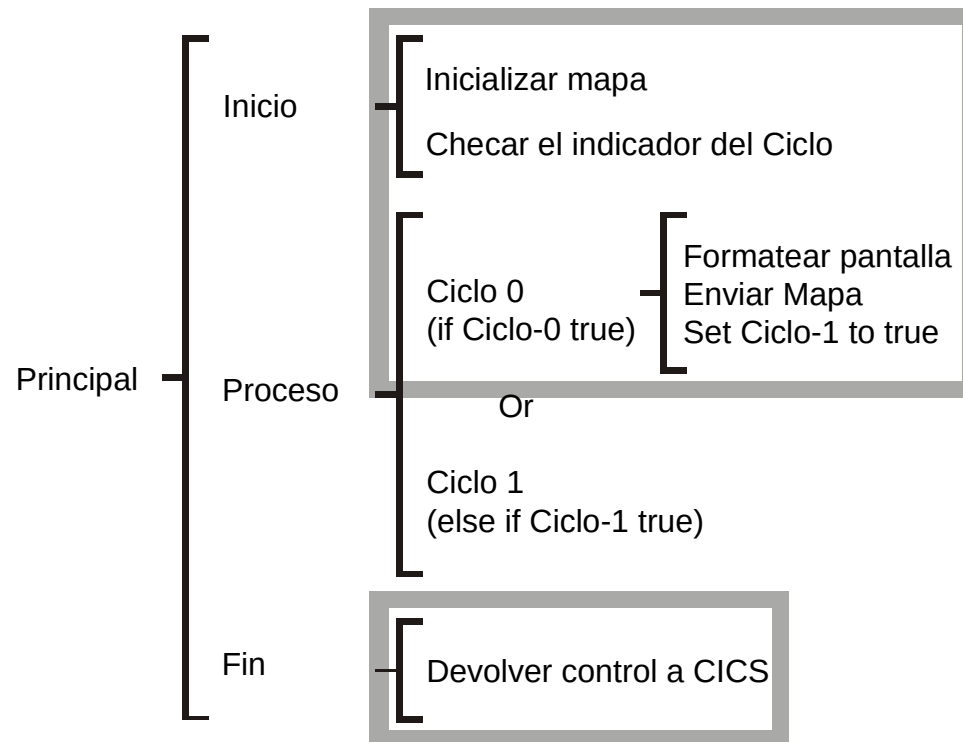
Comunicación con un programa On-Line





Comunicación con un programa On-Line

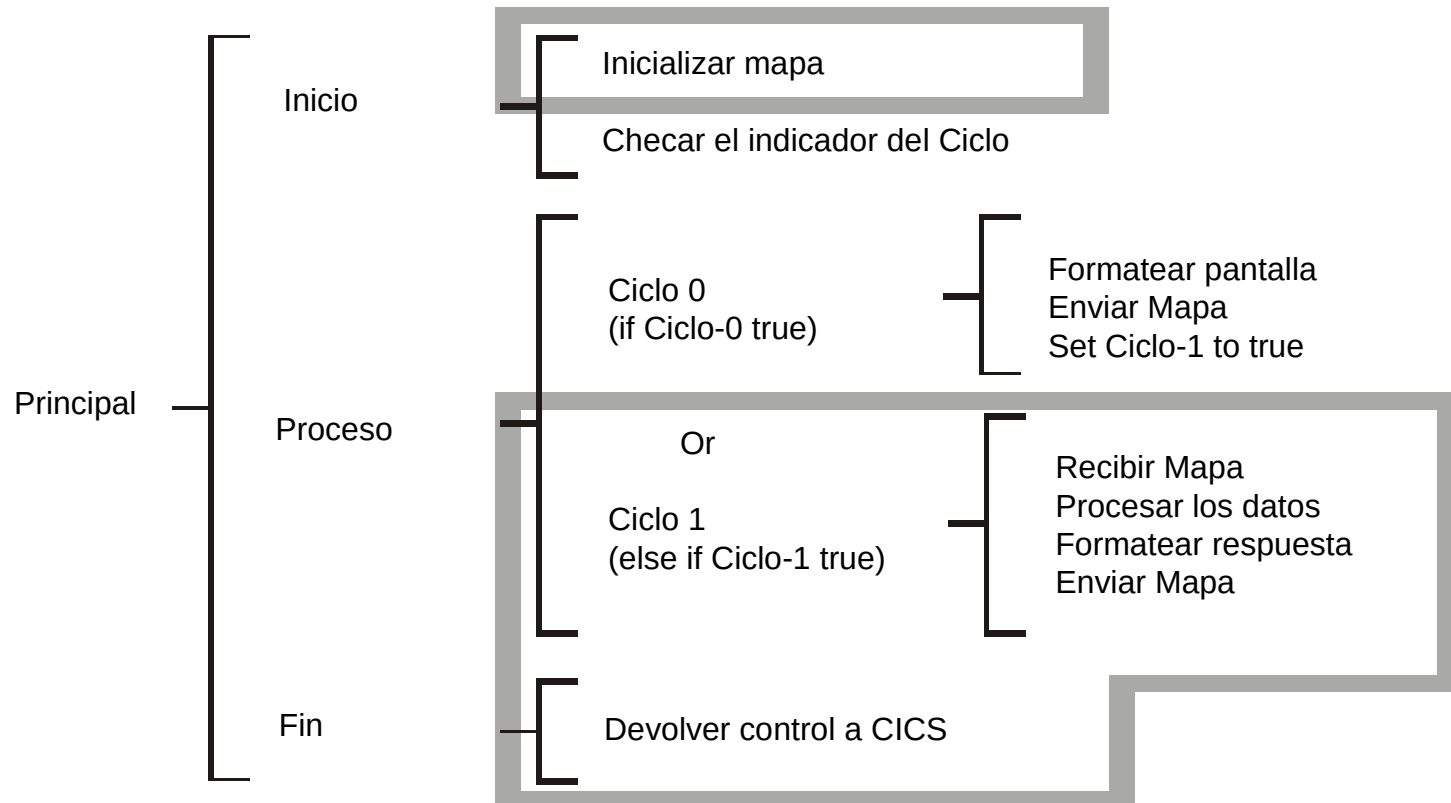
Lógica de un programa CICS (Ciclo 0)





Comunicación con un programa On-Line

Lógica de un programa CICS (Ciclo 1)





Contenido

Introducción a CICS

Comunicación con programas On-Line

Interfase de Programación de Alto Nivel (HLPI)

Mapas

Mandatos para el Control de Terminales

Manejo de archivos VSAM

Browsing y Colas TS

Arquitectura Altamira

Ciclos

Areas de Comunicación

Link

XCTL

Return



Comunicación con un programa On-Line

Áreas de comunicación

Cuando un programa es descargado de la memoria, los valores de la Working-Storage se pierden, por eso se utiliza el área de comunicación para recordar los valores que le permiten pasar datos entre programas.

En un programa CICS hay dos áreas de comunicación utilizados para pasar datos entre programas en ejecución :

- Área de Comunicación de la Working-Storage Section.
- Área de Comunicación de la Linkage Section.

```
DATA DIVISION.  
WORKING-STORAGE SECTION.  
.  
.  
.  
01 WCA-COMMAREA.  
.  
.  
LINKAGE SECTION.  
01 DFHCOMMAREA      PIC X(200).  
  
PROCEDURE DIVISION.
```



Comunicación con un programa On-Line

Restricciones del compilador de COBOL-CICS

- No pueden definirse archivos en los programas de aplicación CICS. En lugar de esto, los archivos se definen en una tabla CICS, la File Control Table (FCT).
- La ENVIRONMENT DIVISION no puede incluir la INPUT-OUTPUT SECTION.
- La DATA DIVISION no puede incluir la FILE SECTION.
- La PROCEDURE DIVISION no puede usar los siguientes verbos de COBOL :

OPEN	CLOSE	READ
WRITE	REWRITE	DELETE
START	READ NEXT	SORT
STOP RUN	ACCEPT	DISPLAY

Comunicación con un programa On-Line

Mandatos para la comunicación entre programas

En CICS, cada uno de los módulos individuales está almacenado como una única unidad en la biblioteca de programas y el CICS se responsabiliza de la unión de los módulos en tiempo de ejecución. De esto se encarga el programa de Control de Programas.

Los programas se ejecutan en varios niveles lógicos :

- Cuando un programa enlaza a otro (LINK), el programa que recibe el control reside en el siguiente nivel lógico inferior.
- Cuando un programa transfiere el control (XCTL) a otro, el programa que recibe el control reside en el mismo nivel lógico, reemplazado al programa que originó la transferencia.
- Cuando un programa pide devolver el control, éste es pasado al siguiente nivel lógico superior.



Contenido

Introducción a CICS

Comunicación con programas On-Line

Interfase de Programación de Alto Nivel (HLPI)

Mapas

Mandatos para el Control de Terminales

Manejo de archivos VSAM

Browsing y Colas TS

Arquitectura Altamira

Ciclos

Areas de Comunicación

Link

XCTL

Return



Comunicación con un programa On-Line

Mandatos para la comunicación entre programas

LINK

El nombre del programa se especifica en la opción PROGRAM. Debe estar definido en la PPT (Tabla de Programas de Proceso) y los datos a transmitir al programa se incluyen en la COMMAREA, utilizando las variables declaradas en la **commarea** de la WORKING-STORAGE SECTION.

Formato :

```
EXEC CICS LINK  
    PROGRAM (nombre del programa)  
    COMMAREA (WCA-COMMAREA)  
END-EXEC
```



Contenido

Introducción a CICS

Comunicación con programas On-Line

Interfase de Programación de Alto Nivel (HLPI)

Mapas

Mandatos para el Control de Terminales

Manejo de archivos VSAM

Browsing y Colas TS

Arquitectura Altamira

Ciclos

Areas de Comunicación

Link

XCTL

Return

Comunicación con un programa On-Line

Mandatos para la comunicación entre programas

XCTL

Cede el control a otro programa en el mismo nivel lógico, sin que exista retorno cuando éste termine, por lo que el programa que realiza el XCTL se libera, descargándose de la memoria.

Formato :

```
EXEC CICS XCTL  
    PROGRAM (nombre del programa)  
    COMMAREA (WCA-COMMAREA)  
END-EXEC
```




Contenido

Introducción a CICS

Comunicación con programas On-Line

Interfase de Programación de Alto Nivel (HLPI)

Mapas

Mandatos para el Control de Terminales

Manejo de archivos VSAM

Browsing y Colas TS

Arquitectura Altamira

Ciclos

Areas de Comunicación

Link

XCTL

Return



Comunicación con un programa On-Line

Mandatos para la comunicación entre programas

RETURN

Devuelve el control al programa de nivel superior, o al CICS al terminar la tarea. Se pueden especificar dos opciones :

- Se puede indicar qué transacción será ejecutada cuando el siguiente **input** llegue del mismo terminal. Así se produce el modo pseudoconversacional.
- Se pueden especificar los datos que se van a pasar a la siguiente transacción.

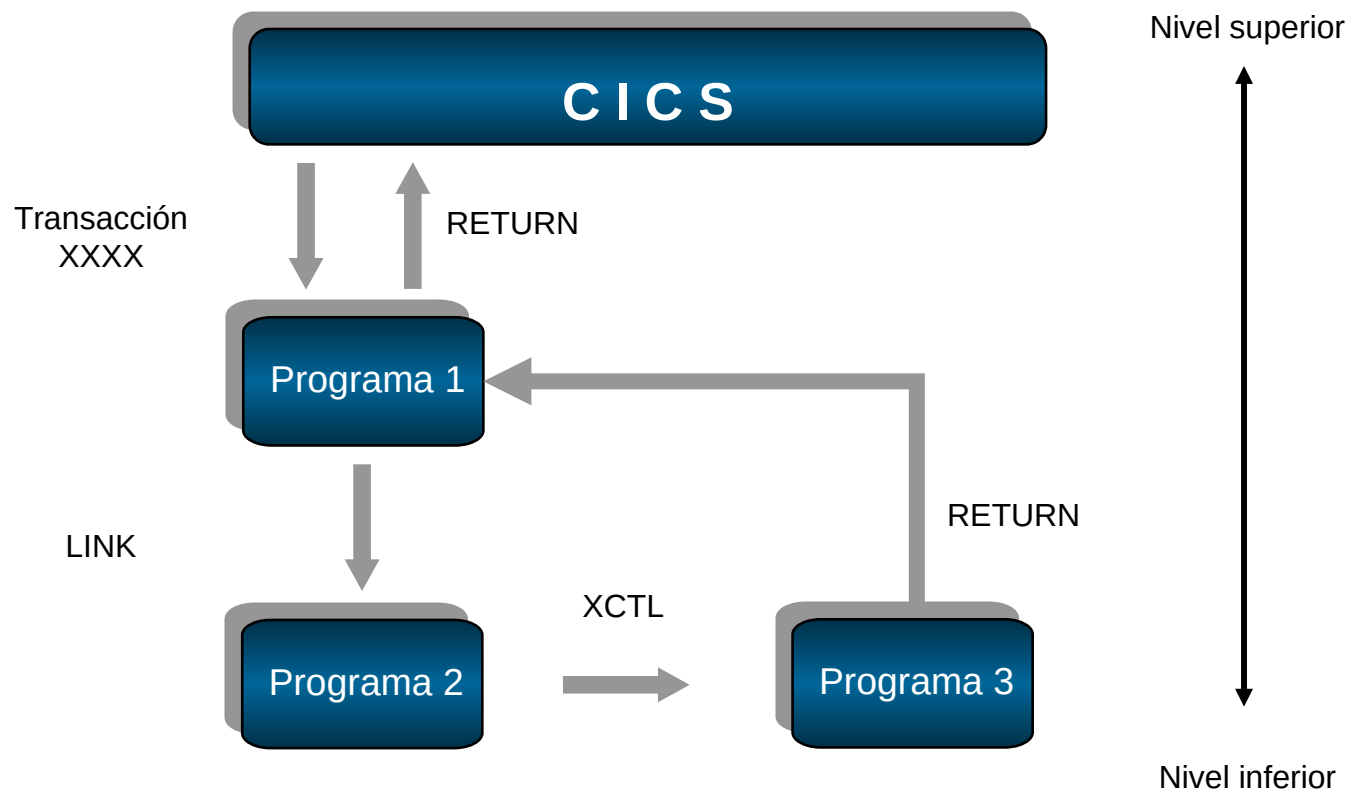
Formato :

```
EXEC CICS RETURN  
      TRANSID (nombre de la transacción)  
      COMMAREA (WCA-COMMAREA)  
END-EXEC
```



Comunicación con un programa On-Line

Mandatos para la comunicación entre programas





Contenido

Introducción a CICS

Comunicación con programas On-Line

Interfase de Programación de Alto Nivel (HLPI)

Mapas

Mandatos para el Control de Terminales

Manejo de archivos VSAM

Browsing y Colas TS

Arquitectura Altamira

**Bloque de Ejecución de la Interfase
(EIB)**

Condiciones de Excepción



Interfase de Programación de Alto Nivel (HLPI)

Esta interfase está compuesta de un **conjunto de mandatos** que se insertan en los programas COBOL cuando se necesita una función del CICS.

De este modo, en un programa CICS habrá Sentencias COBOL y Mandatos CICS.

Bloque EIB

En el traductor de mandatos existen declaraciones que automáticamente van a copiar un **BLOQUE DE EJECUCIÓN DE LA INTERFASE** (EIB) en la LINKAGE SECTION del programa. Los campos del EIB contienen información que puede ser necesaria durante la ejecución del programa.

El programa puede recuperar información del EIB usando el campo apropiado, pero no se puede escribir en el bloque. Además, los nombres de estos campos son palabras reservadas. La información suministrada por el bloque EIB es la siguiente :



Interfase de Programación de Alto Nivel (HLPI)

Bloque EIB

EIBCALEN	Contiene la longitud del área de comunicación (COMMAREA), que ha sido pasada al programa desde el último programa, si no se ha pasado área de comunicación, EIBCALEN valdrá cero. Esto ocurre la primera vez que un programa ha sido ejecutado.
EIBDATE	La fecha en que fue iniciada la tarea. Es fecha juliana. Su formato es "AAAADDD" (<i>PIC S9(7) COMP-3</i>).
EIBFN	Número asociado al último comando de CICS ejecutado. EIBFN ocupa 2 bytes y el número se guarda en formato binario.
EIBTIME	Hora con formato "0HHMMSS+". (<i>PIC S9(7) COMP-3</i>).
EIBTRMID	Nombre de la terminal asociada a la tarea.
EIBRESP	Código de retorno del último comando CICS ejecutado.



Interfase de Programación de Alto Nivel (HLPI)

Bloque EIB

EIBAID	Guarda la tecla de control utilizada para transmitir la última entrada de datos por pantalla. La definición de las teclas de control se realiza con la copy DFHAID . La lista estándar de dichas teclas es la siguiente :	
	Nombre Simbólico	Función
	DFHENTER	ENTER
	DFHCLEAR	CLEAR
	DFHPPF1	PF1
	DFHPPF2	PF2
	DFHPPF3	PF3



Contenido

Introducción a CICS

Comunicación con programas On-Line

Interfase de Programación de Alto Nivel (HLPI)

Mapas

Mandatos para el Control de Terminales

Manejo de archivos VSAM

Browsing y Colas TS

Arquitectura Altamira

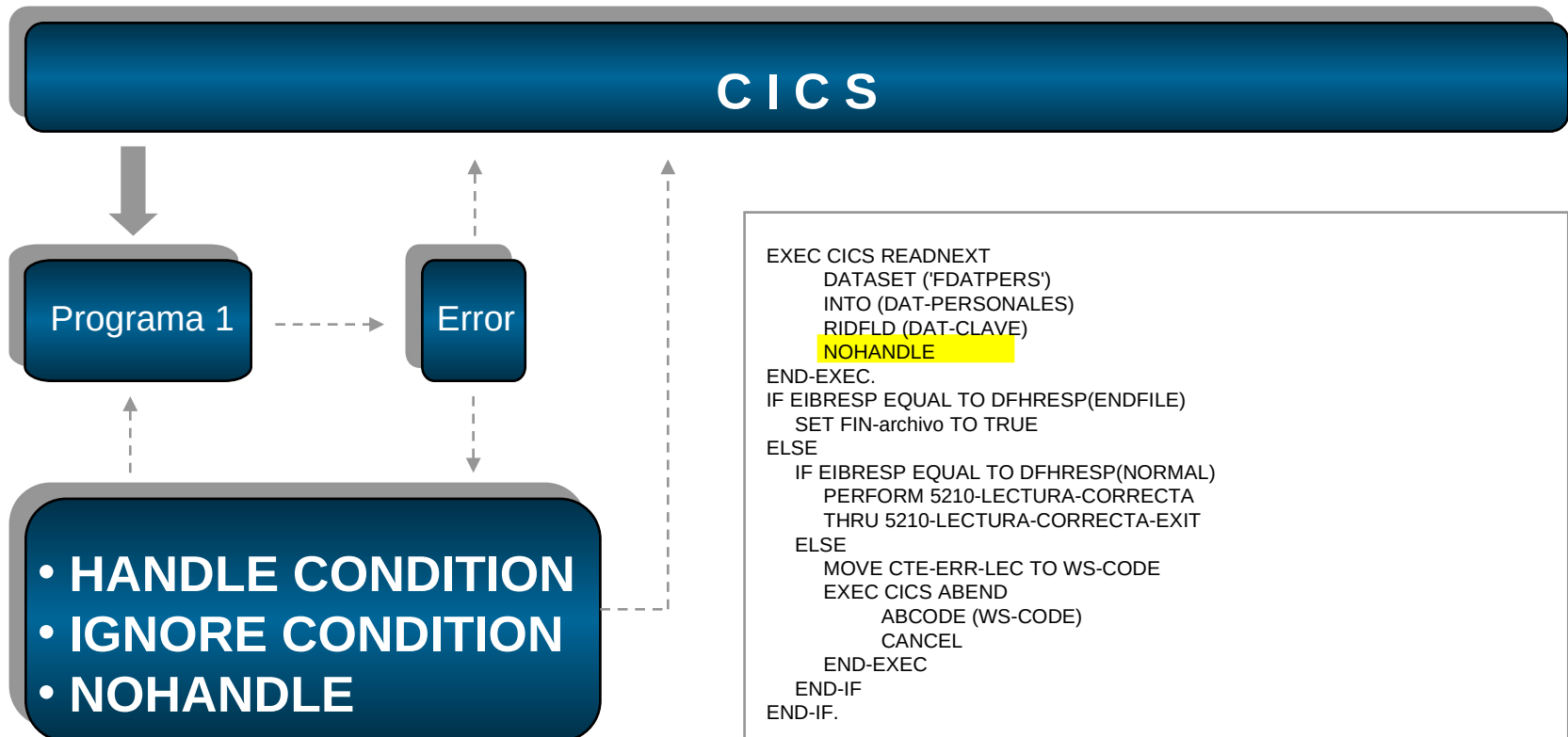
Bloque de Ejecución de la Interfase (EIB)

Condiciones de Excepción



Interfase de Programación de Alto Nivel (HLPI)

Condiciones de excepción





Contenido

Introducción a CICS

Comunicación con programas On-Line

Interfase de Programación de Alto Nivel (HLPI)

Mapas

Mandatos para el Control de Terminales

Manejo de archivos VSAM

Browsing y Colas TS

Arquitectura Altamira

Introducción

Mapas Físicos

Mapas Simbólicos

Mandatos para el control de terminales



Basic Mapping Support (BMS)

La utilidad BMS permite realizar y gestionar **mapas** (pantallas) de CICS. BMS suministra servicios de transformación de datos para ayudar a preparar las corrientes de salida e interpretar las corrientes de entrada de la red de terminales.

Componentes de las operaciones del BMS

La situación de los campos en la corriente de datos los da el BMS, a través de las *Tablas de Formatos de Datos*, llamadas "mapas". Los mapas se crean mediante macros BMS que se codifican en ensamblador y que contienen información sobre :

- ☐ Posiciones de los campos.
- ☐ Atributos.
- ☐ Datos de cabecera e iniciales.



Basic Mapping Support (BMS)

Un mapa BMS está constituido por un conjunto de instrucciones y comandos que dan formato a los campos, tiene dos formas, una denominada **MAPA FÍSICO** y otra **MAPA SIMBÓLICO**.

Mapa Físico	Define y da nombre a los campos que se pueden escribir o recibir, desde los terminales. Es codificado por el programador mediante macros de ensamblador.
Mapa Simbólico ó Lógico	<p>Es generado con las mismas macros que el Mapa Físico con el que está asociado. Suministra los nombres simbólicos para los campos definidos en el mapa físico, que se utilizan en el programa.</p> <p>Es generado al compilar el mapa físico por el compilador de mapas.</p>



Basic Mapping Support (BMS)

Compilación del MAPA y generación del COPY.

Mapa Físico

```
MSET01 DFHMSD TYPE=MAP,      -
          MODE=INOUT,        -
          LANG=COBOL
MAPA01 DFHMDI SIZE=(24,80)
          DFHMDF POS=(1,1),   -
          LENGTH=8,          -
          INITIAL='Nombre :'
NAME    DFHMDF POS=(1,10),   -
          LENGTH=20
          DFHMDF POS=(1,21),  -
          ATTRB=ASKIP
          DFHMSD TYPE=FINAL
          END
```

Mapa Simbólico

```
01 MAPA01I.
    02 FILLER                PIC X(12).
    02 NAMEL                PIC S9(4) COMP.
    02 NAMEF                PIC X.
    02 NAMEA                PIC X.
    02 NAMEI                PIC X(20).
01 MAPA01O REDEFINES MAPA01I.
    02 FILLER                PIC X(12).
    02 NAMEO                PIC X(20).
```



Contenido

Introducción a CICS

Comunicación con programas On-Line

Interfase de Programación de Alto Nivel (HLPI)

Mapas

Mandatos para el Control de Terminales

Manejo de archivos VSAM

Browsing y Colas TS

Arquitectura Altamira

Introducción

Mapas Físicos

Mapas Simbólicos

Mandatos para el control de terminales



Basic Mapping Support (BMS)

Macros para la generación de Mapas Físicos - DFHMSD

DFHMSD XXXX=YYYY

Ejemplo :

```
MAPSET DFHMSD TYPE=MAP,
        MODE=INOUT,
        LANG=COBOL
```

Un MAPSET, es un conjunto de uno ó más mapas que el CICS considera como una unidad de gestión, en esta macro se define tanto el nombre del mapa como los siguientes parámetros :

mapset	Se declara el nombre del mapa, el cual es obligatorio, debe ser único y tener de 1 a 7 caracteres.
TYPE	En esta parte se especifica que se trata de un mapa.
MODE	Indica si es un mapa de in, out o inout.
LANG	Especifica el lenguaje en el que se quiere que se genere la COPY del mapa simbólico.



Basic Mapping Support (BMS)

Macros para la generación de Mapas Físicos - DFHMSD

<p>DFHMSD XXXX=YYYY</p> <p>Ejemplo :</p> <p>DFHMSD TYPE=FINAL END</p>	<p>Indica que ya no siguen más elementos en la definición del mapa.</p>
---	---



Basic Mapping Support (BMS)

Macros para la generación de Mapas Físicos - DFHMDI

DFHMDI XXXX=YYYY

Ejemplo :

```
MAP      DFHMDI SIZE=(24,80)
          LINE=1,
          COLUMN=1
```

En esta macro da el nombre del mapa simbólico, así como sus características como son :

- Tamaño
- Posición
- Valores por defecto para este mapa

Normalmente se definirá la pantalla completa (24 x 80) comenzando en la línea 1 columna 1.

SIZE	Dimensión del mapa en longitud y anchura.
LINE	Línea en la que comenzará el mapa.
COLUMN	Columna en la que comenzará el mapa.



Basic Mapping Support (BMS)

Macros para la generación de Mapas Físicos - DFHMDF

DFHMDF XXXX=YYYY

Ejemplo :

DFHMDF POS=(10,30)

En esta macro se definen los campos referenciables y no referenciables en el programa, así como su posición, longitud, atributos, etc. en el terminal.

Para cada definición de campo se necesita una macro DFHMDF.

Un campo referenciable permitirá al programa enviar o recibir datos desde/hacia ese campo.

En ese caso la macro DFHMDF debe tener un nombre asociado, el cual será el nombre de ese campo.

Debe haber una definición para cada campo a manejar en la pantalla, sólo es necesario el nombre del campo si va a ser referenciado desde el programa.



Basic Mapping Support (BMS)

Macros para la generación de Mapas Físicos - DFHMDF

DFHMDF XXXX=YYYY	<div>Parámetros</div> <table> <tr> <td data-bbox="666 604 904 761">POS</td><td data-bbox="904 604 1767 761">Especifica la posición de una variable o literal en pantalla.</td></tr> <tr> <td data-bbox="666 761 904 918">LENGHT</td><td data-bbox="904 761 1767 918">Se usa para especificar la longitud de un campo.</td></tr> <tr> <td data-bbox="666 918 904 1075">IC</td><td data-bbox="904 918 1767 1075">Pone el cursor en la primera posición del campo.</td></tr> <tr> <td data-bbox="666 1075 904 1275">INITIAL</td><td data-bbox="904 1075 1767 1275">Establece el contenido inicial de este campo.</td></tr> </table>	POS	Especifica la posición de una variable o literal en pantalla.	LENGHT	Se usa para especificar la longitud de un campo.	IC	Pone el cursor en la primera posición del campo.	INITIAL	Establece el contenido inicial de este campo.
POS	Especifica la posición de una variable o literal en pantalla.								
LENGHT	Se usa para especificar la longitud de un campo.								
IC	Pone el cursor en la primera posición del campo.								
INITIAL	Establece el contenido inicial de este campo.								



Basic Mapping Support (BMS)

Macros para la generación de Mapas Físicos - DFHMDF

DFHMDF XXXX=YYYY

Parámetros *Continuación*

ATTRB	Define las características del campo respecto a protección, contenido, luminosidad, inserción del cursor, o campo modificado.	
	ASKIP	Salto al siguiente campo.
	PROT	Protegido.
	UNPROT	Desprotegido.



Basic Mapping Support (BMS)

Macros para la generación de Mapas Físicos - DFHMDF

DFHMDF XXXX=YYYY

Ejemplo :

```
DFHMDF POS=( 1,1), -
  ATTRB=(ASKIP,BRT), -
  LENGTH=8, -
  INITIAL='Nombre: ' -
NAME DFHMDF POS=( 1,76), -
  ATTRB=(ASKIP,BRT) -
  LENGTH= 20, -
```

Parámetros *Continuación*

ATTRB

Define las características del campo respecto a protección, contenido, luminosidad, inserción del cursor, o campo modificado.

NUM

Bloqueo numérico.

BRT

Intensidad brillante.

NORM

Intensidad normal.



Contenido

Introducción a CICS

Comunicación con programas On-Line

Interfase de Programación de Alto Nivel (HLPI)

Mapas

Mandatos para el Control de Terminales

Manejo de archivos VSAM

Browsing y Colas TS

Arquitectura Altamira

Introducción

Mapas Físicos

Mapas Simbólicos

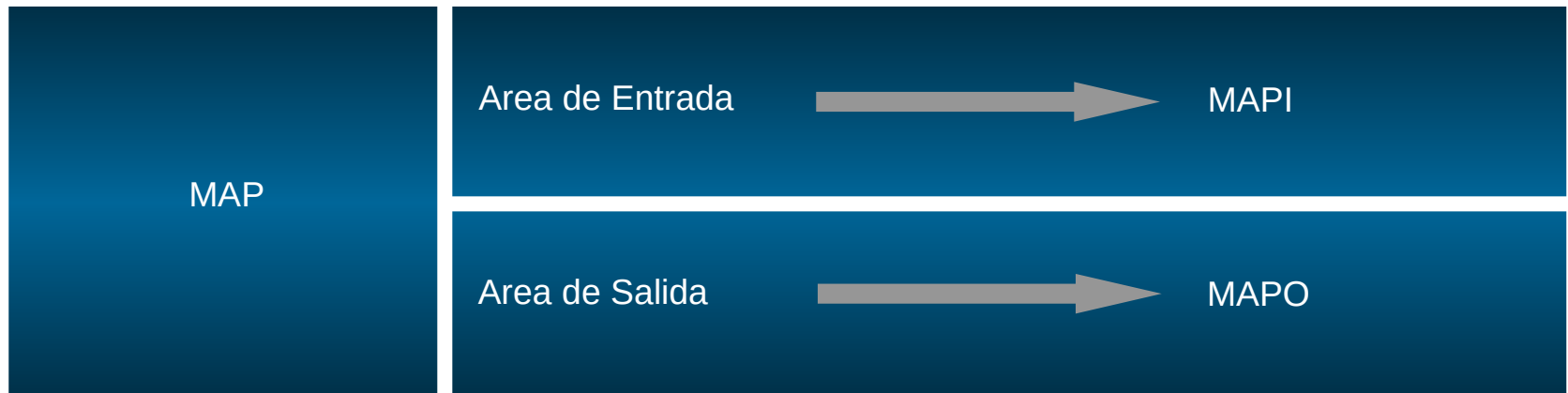
Mandatos para el control de terminales



Basic Mapping Support (BMS)

Macros para la generación de Mapas Simbólicos

Compilación y generación del **COPY**





Basic Mapping Support (BMS)

Macros para la generación de Mapas Simbólicos

Compilación y generación del **COPY**

L (longitud)	Sirve para posicionar el cursor en un campo específico.
F (flag)	Indica si el campo ha sido modificado.
I (input)	Contiene el dato introducido por el usuario y recibido por el programa.
A (atributo)	Modifica desde el programa los atributos de los campos.
O (output)	Contiene el dato procesado por el programa y que el BMS va a enviar al terminal.



Contenido

Introducción a CICS

Comunicación con programas On-Line

Interfase de Programación de Alto Nivel (HLPI)

Mapas

Mandatos para el Control de Terminales

Manejo de archivos VSAM

Browsing y Colas TS

Arquitectura Altamira

Introducción

Mapas Físicos

Mapas Simbólicos

Mandatos para el control de terminales



Basic Mapping Support (BMS)

Mandatos para el Control de Terminales

Los mandatos de control de terminales permiten invocar desde el programa COBOL al BMS para enviar y recibir mapas. Por tanto, estos mandatos permiten comunicarse directamente al usuario, que introduce datos desde una terminal, con el programa.

SEND MAP

Se utiliza para suministrar la transformación de datos de salida hacia la terminal.

La sintaxis es :

```
EXEC CICS SEND
  MAP      ('Nombre del mapa')
  MAPSET   ('Nombre del MAPSET')
  FROM     (Area de datos 'OUTPUT' del mapa
            simbólico)
END-EXEC.
```



Basic Mapping Support (BMS)

Mandatos para el Control de Terminales

RECEIVE MAP

Recibe una entrada de una terminal. Se usa para transformar datos en un área definida por el programa.

Una vez transformados se pueden referenciar los campos mediante los especificados en la macro.

LA CONDICIÓN MAPFAIL

La condición de MAPFAIL se produce si los datos recibidos están no formateados o tienen longitud cero ó si el operador pulsó la tecla CLEAR o ENTER o si pulsó una tecla PF sin haber introducido ningún dato en la terminal.

```
EXEC CICS RECEIVE
  MAP      ('Nombre del mapa')
  MAPSET   ('Nombre del MAPSET')
  INTO     (Area de datos 'INPUT' del mapa
            simbólico)
END-EXEC.
```



Mandatos para el Control de Terminales

RECEIVE MAP

Recibe una entrada de una terminal. Se usa para transformar datos en un área definida por el programa.

Una vez transformados se pueden referenciar los campos mediante los especificados en la macro.

LA CONDICIÓN MAPFAIL

La condición de MAPFAIL se produce si los datos recibidos están **no formateados** o tienen longitud cero ó si el operador pulsó la tecla CLEAR o ENTER o si pulsó una tecla PF sin haber introducido ningún dato en la terminal.

```
EXEC CICS RECEIVE
  MAP      ('Nombre del mapa')
  MAPSET   ('Nombre del MAPSET')
  INTO     (Area de datos 'INPUT' del mapa
            simbólico)
END-EXEC.
```

Contenido

Introducción a CICS

Comunicación con programas On-Line

Interfase de Programación de Alto Nivel (HLPI)

Mapas

Mandatos para el Control de Terminales

Manejo de archivos VSAM

Browsing y Colas TS

Arquitectura Altamira

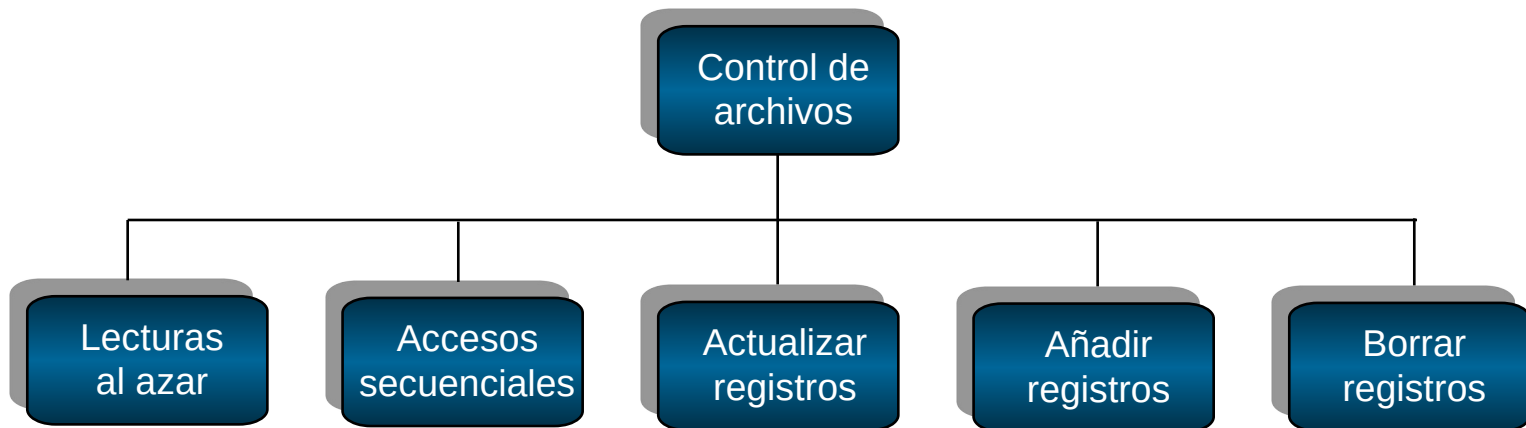


Control de archivos VSAM

Las peticiones de registros de datos de un archivo o conjunto de datos desde un programa son tratados siempre por el CICS.

Las características de los archivos o conjuntos de datos se describen en la Tabla de Control de archivos, por lo que el programa lo único que necesita es pedir un registro de un determinado archivo.

Las facilidades que proporciona el Control de archivos para una Aplicación son :





Control de archivos VSAM (READ)

CICS proporciona los comandos Input/Output, READ y WRITE para el acceso a archivos.

Estos se diferencian de los verbos COBOL en que es el propio CICS quien interactúa con los archivos.

Como parte del comando READ, CICS busca, localiza y deposita el registro buscado en el Record Layout de la Working Storage Section.

```
EXEC CICS READ
      DATASET ('Nombre Archivo')
      INTO      (WS-RECORD)
      RIDFLD    (WS-CLAVE)
      option...
END-EXEC.
```

DATASET	Identifica el nombre del archivo definido en la FCT.
INTO	Identifica la variable de la Working Storage Section que contiene el layout del registro.
RIDFLD	Identifica el campo clave, en la descripción del registro, dentro de la Working Storage Section.
Option	UPDATE (queda bloqueado el registro para ser actualizado).
	EQUAL (trae el registro que sea igual al especificado).
	GTEQ (trae el primer registro cuya clave sea => a la especificada).



Control de archivos VSAM (REWRITE, WRITE y DELETE)

EXEC CICS REWRITE

 DATASET ('Nombre Archivo')

 FROM (WS-RECORD)

END-EXEC.

Este comando actualiza el registro que se acaba de leer.

Sólo se puede ejecutar si previamente se ha ejecutado el comando READ con la opción UPDATE.

EXEC CICS WRITE

 DATASET ('Nombre Archivo')

 FROM (WS-RECORD)

 RIDFLD (WS-CLAVE)

END-EXEC.

Este comando añade un nuevo registro al archivo VSAM.

EXEC CICS DELETE

 DATASET ('Nombre Archivo')

 RIDFLD (WS-CLAVE)

END-EXEC.

Este comando borra un registro del archivo VSAM.



Contenido

Introducción a CICS

Comunicación con programas On-Line

Interfase de Programación de Alto Nivel (HLPI)

Mapas

Mandatos para el Control de Terminales

Manejo de archivos VSAM

Browsing y Colas TS

Arquitectura Altamira



Browsing y Colas TS

Operaciones Browsing en archivos VSAM

El término browsing se utiliza para referirse a una búsqueda secuencial de registros iniciada a partir de un punto concreto.

Para comenzar el proceso, primero debe realizarse un posicionamiento en un registro en específico y a continuación realizar la lectura secuencial del archivo.

Para realizar el proceso se debe suministrar una serie de mandatos característicos del browsing que configuran tres etapas :

- ✓ Establecimiento del entorno browsing (**STARTBR**)
- ✓ Recuperación de un registro (**READNEXT**)
- ✓ Final del Browsing (**ENDBR**)



Comandos Browsing en archivos VSAM (SRTARTBR, READNEXT, ENDBR)

<pre>EXEC CICS STARTBR DATASET ('Nombre Archivo') RIDFLD (WS-CLAVE) GTEQ NOHANDLE END-EXEC.</pre>	<p>Indica al CICS el registro desde el que se desea comenzar a leer en el archivo.</p> <p>GTEQ, Significa que se desea obtener el primer registro cuya clave sea mayor o igual que la especificada.</p>
<pre>EXEC CICS READNEXT DATASET ('Nombre Archivo') INTO ('WS-RECORD') RIDFLD (WS-CLAVE) NOHANDLE END-EXEC.</pre>	<p>READNEXT, obtiene el primer y los sucesivos registros siguientes de browse, es decir, en orden secuencial ascendente.</p> <p>READPREV, obtiene el primer y los anteriores registros browse, es decir, en orden secuencial descendente.</p>
<pre>EXEC CICS ENDBR DATASET ('Nombre Archivo') NOHANDLE END-EXEC.</pre>	<p>Este mandato finaliza un browsing liberando todos los recursos que el CICS asoció al mismo.</p> <p>Un browsing retiene muchos recursos del sistema, es importante liberar estos recursos cuanto antes.</p>



Colas TS (Gestión de Memoria Temporal)

La información que gestionan se almacena temporalmente en memoria, por eso es volátil y no se pueden borrar físicamente, pues no existen registros como tales , aunque lo que sí se puede hacer es marcarlos.

Los datos temporales son guardados como registros (items), de longitud variable, con unos nombres simbólicos. Son colas secuenciales.

No es necesario definir en ninguna tabla los nombres simbólicos de las colas TS, sólo sería necesario definirlos en el caso de querer que fueran recuperables al arrancar el CICS para poder ser seleccionado por la tarea.

Las colas TS se pueden recuperar más de una vez. Su item, puede leerse tantas veces como se quiera. Existe un único índice de lectura y otro de grabación a nivel CICS.



Colas TS (Gestión de Memoria Temporal)

Escritura, lectura y borrado de registros en colas TS

EXEC CICS WRITEQ TS QUEUE (Nombre de cola) FROM (WS-RECORD) ITEM (WS-ITEM) NOHANDLE END-EXEC.	Comando que permite agregar nuevos registros a la cola, o crearla en caso de que no exista.	
	QUEUE	Especifica el nombre de la cola donde los datos se almacenan.
	FROM	Indica el área de datos que se desea grabar en la cola.
	ITEM	El programa guarda el número asignado al elemento (ítem) escrito.
EXEC CICS READQ TS QUEUE (Nombre de cola) INTO (WS-RECORD) ITEM (WS-ITEM) NOHANDLE END-EXEC.	Comando utilizado para acceder a la información contenida en la cola TS.	
	INTO	Este parámetro especifica el área definida por el usuario en la que se van a recuperar los datos.
EXEC CICS DELETEQ TS QUEUE (Nombre de cola) NOHANDLE END-EXEC.	Comando utilizado para borrar la información contenida en la cola TS.	

Contenido

Introducción a CICS

Comunicación con programas On-Line

Interfase de Programación de Alto Nivel (HLPI)

Mapas

Mandatos para el Control de Terminales

Manejo de archivos VSAM

Browsing y Colas TS

Arquitectura Altamira



Arquitectura Altamira

Definición

Arquitectura Altamira se encarga de proveer funciones comunes para los distintos módulos aplicativos con el fin de simplificar la lógica de los mismos facilitando de esta manera su desarrollo y su mantenimiento.

Características

- ❑ Lógica de programación repetitiva, absorbida por la arquitectura, minimizando el esfuerzo del desarrollo :
 - Emisión / Recepción de mensajes de terminal
 - Gestión de condiciones de excepción (errores, avisos, ABENDS).
- ❑ Seguridad contra accesos no autorizados.
- ❑ Aislamiento del programa de aplicación de la mayor parte de los accesos al monitor comunicaciones (CICS).
- ❑ Estandarización.



Arquitectura Altamira

Características (continuación)

- ❑ Soporta todas las transacciones de un teleproceso bancario, independientemente del tipo de terminal conectado.
- ❑ Facilita la incorporación de nuevas aplicaciones.
- ❑ Parametrización.
- ❑ Facilita el mantenimiento de aplicaciones.
- ❑ Las principales características del módulo de Arquitectura son :
 - ✓ Protege la inversión en equipamiento y en lógica aplicativa.
 - ✓ Realiza diversas funciones comunes del sistema.



Arquitectura Altamira

Las principales funciones comunes que realiza la Arquitectura son :

- ☐ Seguridad de acceso a las funciones “on-line” del sistema.
- ☐ Optimización del uso de telecomunicaciones.
- ☐ Interacción con el monitor de teleprocesamiento.
- ☐ Distribución de informes.
- ☐ Administración de totales de cajeros.
- ☐ Mantenimiento de un journal de transacciones monetarias (diario de movimientos).
- ☐ Mantenimiento de los registros de control del sistema.

Arquitectura Altamira

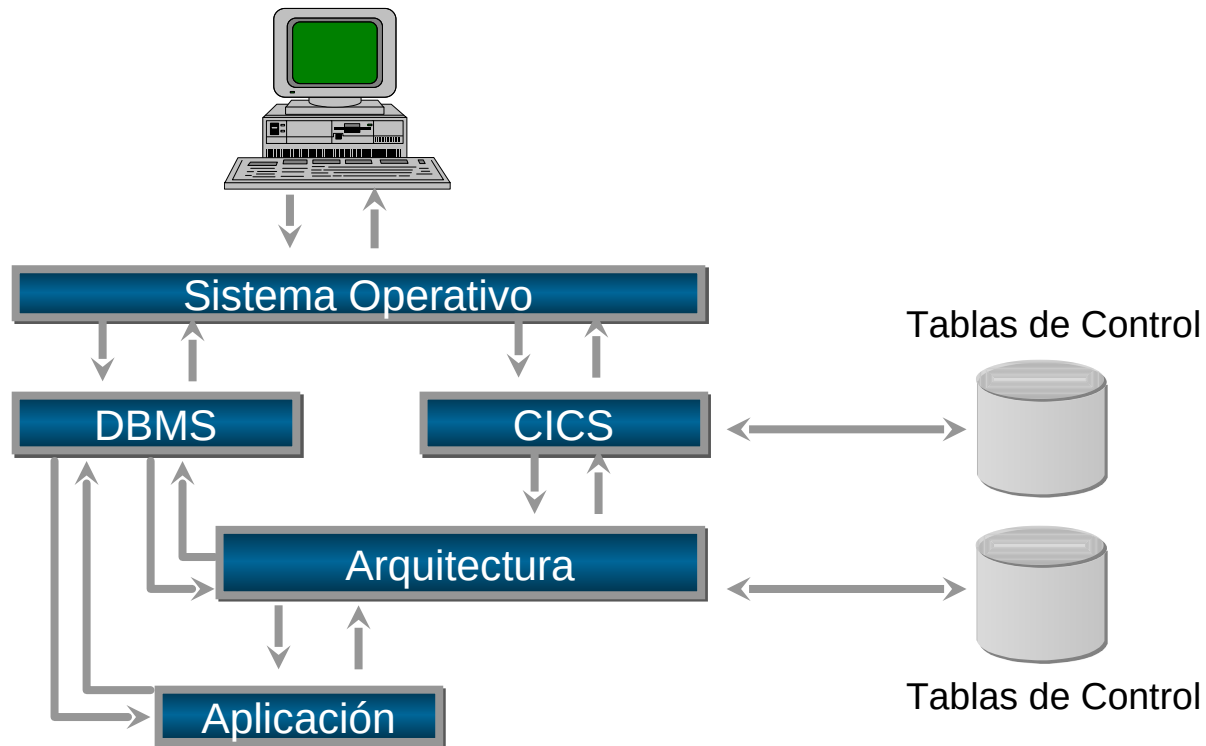
Principales funciones (continuación)

- ☐ En su operatoria “on-line”, Altamira permite el uso de los siguientes tipos de terminales :
 - ✓ Terminales inteligentes (PC's o PS's) en red local.
 - ✓ Terminales no inteligentes.
 - ✓ Otras terminales, utilizando los manejadores de dispositivos específicos (cajeros automáticos, terminales de punto de venta, etc.).
 - ✓ Telediscos.
- ☐ Gestionar impresión de documentos, libreta, banda y diario magnético.
- ☐ Checar el estado de las bases de datos.
- ☐ Control de autorizaciones.
- ☐ Efectuar cambios de transacción (switch de transacción).



Arquitectura Altamira

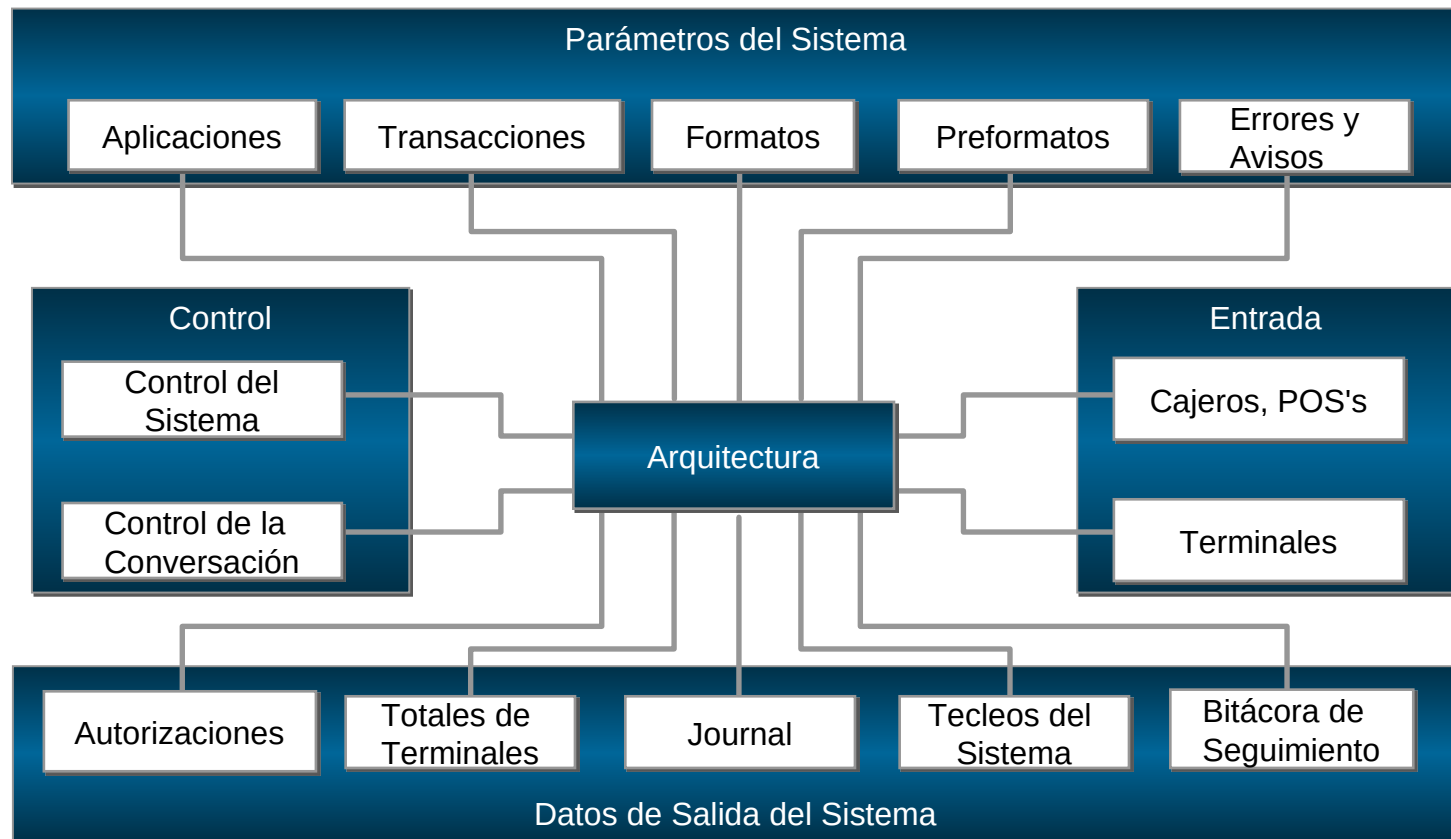
Núcleo central responsable directo de la gestión del entorno técnico en Altamira.





Arquitectura Altamira

Esquema de funcionamiento de la Arquitectura Altamira.





Arquitectura Altamira

Area de Comunicación con la Arquitectura

Es el área básica donde se comunican las aplicaciones con la Arquitectura denominada CAA (Commarea de Arquitectura de Aplicaciones).

El área de comunicación con la arquitectura (CAA) es utilizada para el diálogo entre los programas de aplicación y la arquitectura.

Mediante esta commarea, la arquitectura informa a las aplicaciones de los parámetros del sistema necesarios para el desarrollo de sus procesos on-line.

Los programas de aplicación utilizan la commarea para realizar peticiones de salida, mensajes, e informan del resultado de los procesos realizados.



Arquitectura Altamira

Formato - Es el conjunto de características de cada uno de los mensajes que viajan entre el Host y los dispositivos locales en oficinas (terminal, impresora, dispensador, etc.).

Preformato - Contiene la parte fija (literales fijos) de un mensaje.

Errores y Avisos - Son dos tipos de mensajes a pantalla que informan al operador sobre algún tipo de incidencia que se haya producido durante el proceso.

Totales - Son conceptos que se utilizan contablemente a nivel de terminal para sumarizar y cuadrar el debe y el haber dentro y fuera de caja.

Journal - Diario de los movimientos contables que se producen en la entidad.

Tecleos - Conjunto de operaciones que se efectúan desde los terminales, donde quedan registradas todas las transacciones que se ejecuta a través de la arquitectura.

Arquitectura Altamira

Cambio de Sesión

Proceso que se produce al cierre del día contable, y en el que:

- ✓ Se cambia la fecha contable del día.
- ✓ Se inicializan las tablas por la siguiente sesión del on-line.
- ✓ Se hace el proceso flip-flop de las tablas que tienen varias versiones.
- ✓ Se generan informes sobre los procesos de la última sesión.

Seguridad

Protección de los diferentes recursos manejados por la arquitectura.

Arquitectura Altamira

Arquitectura Central

- ☐ Gestiona la entrada de cada Transacción al Sistema
- ☐ Controla todas las funciones comunes
- ☐ Identifica el tipo de Aplicación
 - Nativa
 - Mixta
 - Extendida
- ☐ Realiza el Formateo / Deformateo de Mapas
- ☐ Programas : QC1CENT y QC!CSAL
- ☐ Control de Comunicación entre las Transacciones Aplicativas
- ☐ Mantenimiento de Arquitectura



Arquitectura Altamira

Arquitectura General

☐ Gestiona el manejo de rutinas internas

- Fechas
- Entidades
- Errores DB2
- Colas TS