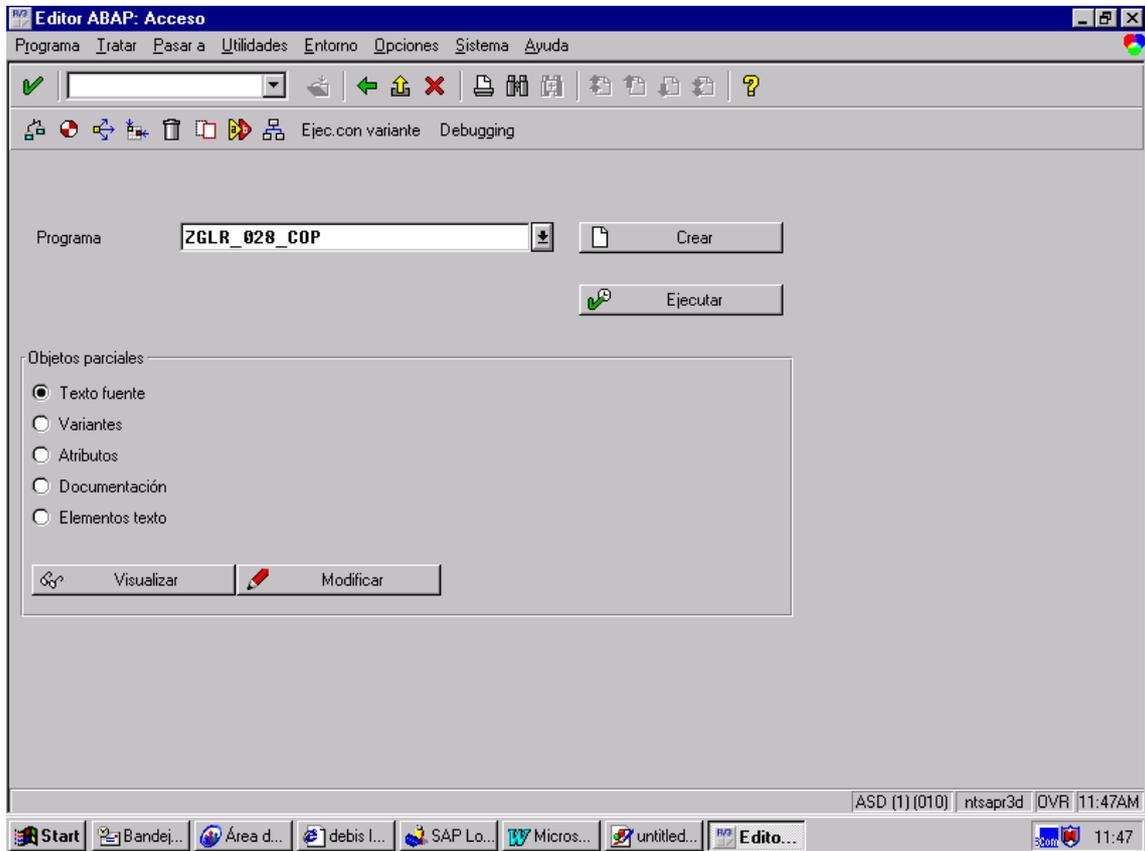


TRANSACCIÓN → SE38

HERRAMIENTA → WORKBENCH ABAP →

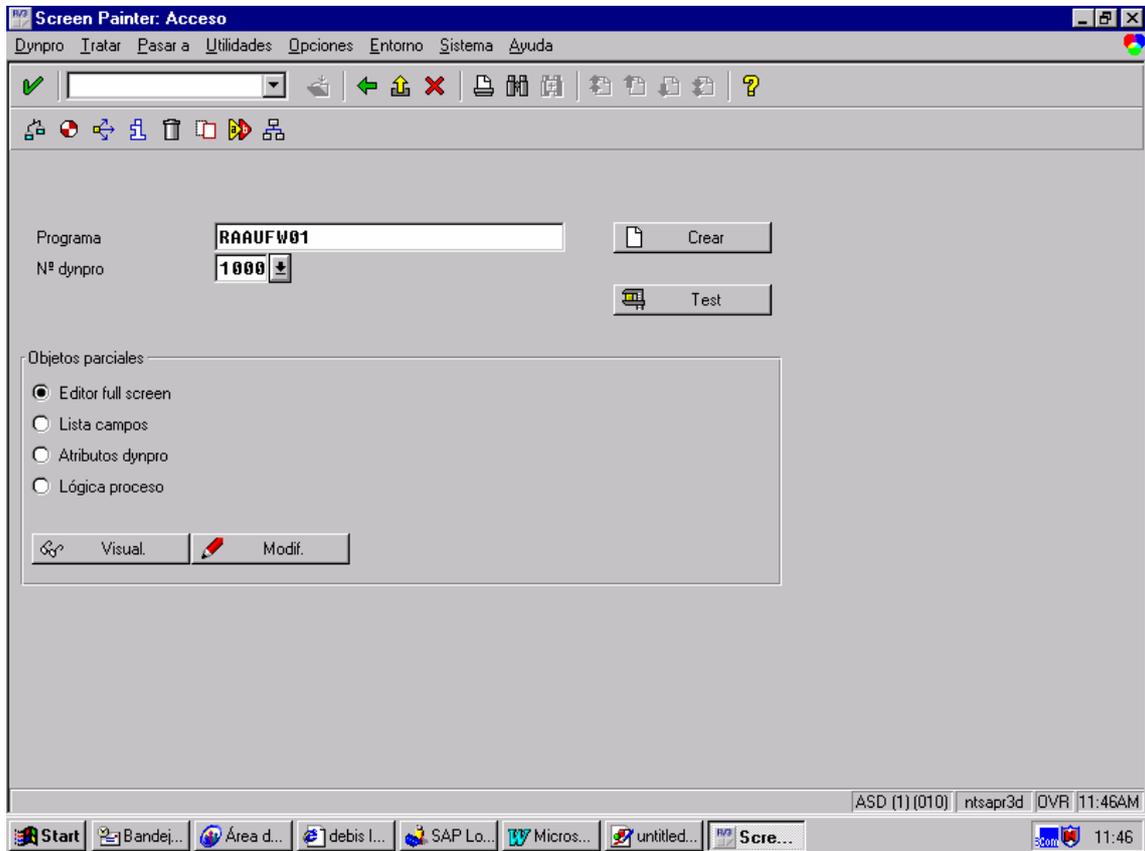
Editor ABAP



TRANSACCIÓN → SE51

HERRAMIENTA → WORKBENCH ABAP →

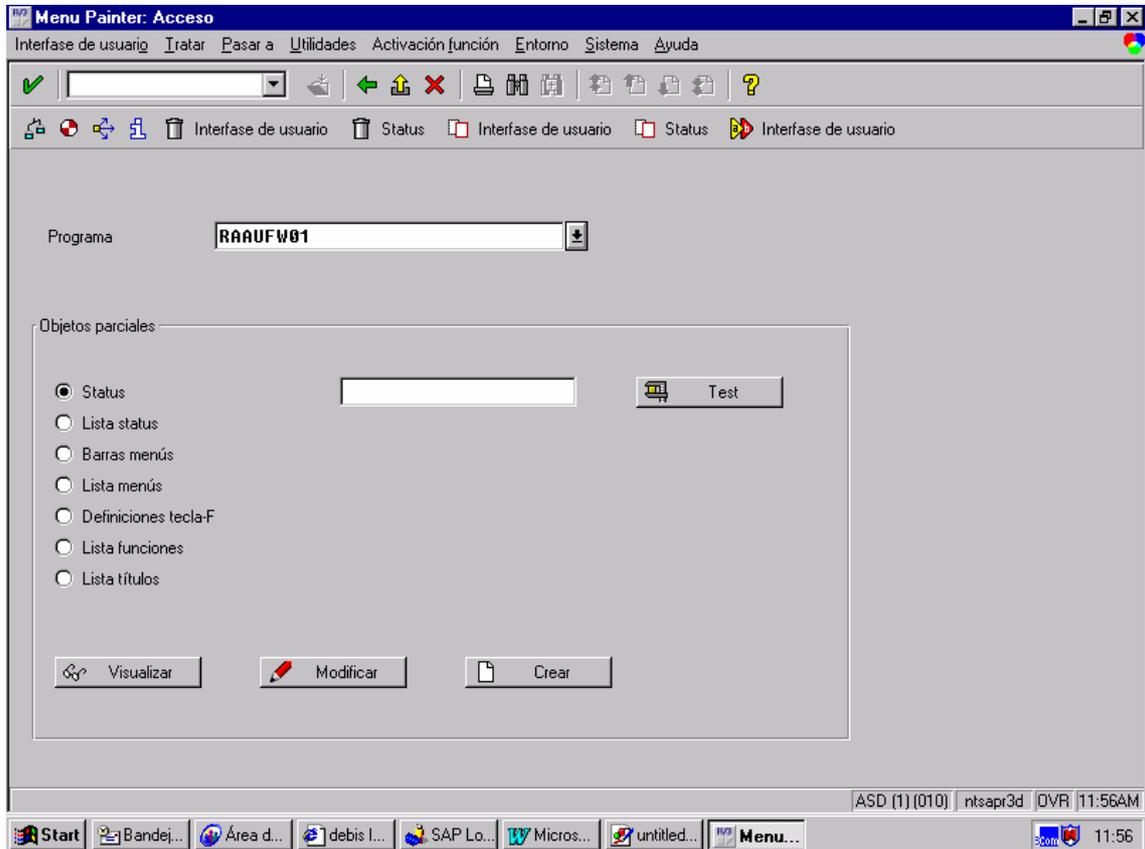
Screen Painter



TRANSACCIÓN → SE41

HERRAMIENTA → WORKBENCH ABAP →

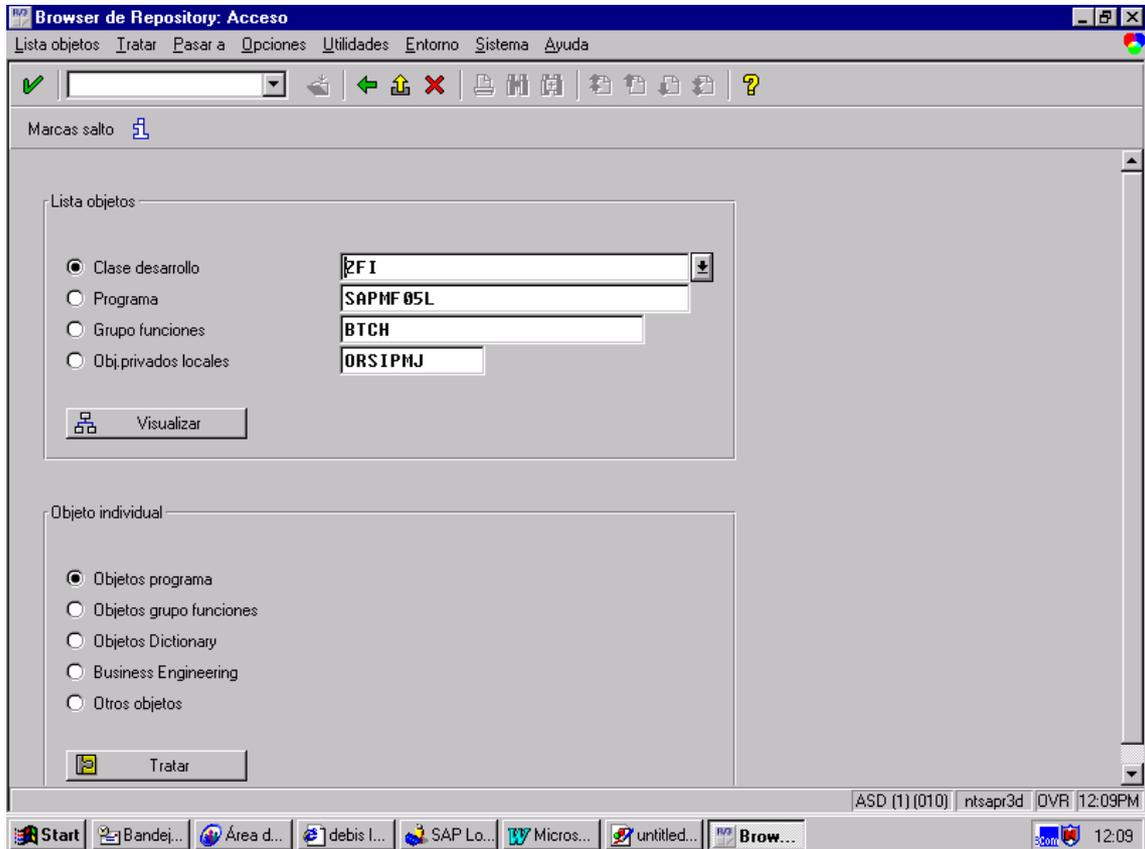
Menu Painter



TRANSACCIÓN → SE80

HERRAMIENTA → WORKBENCH ABAP →

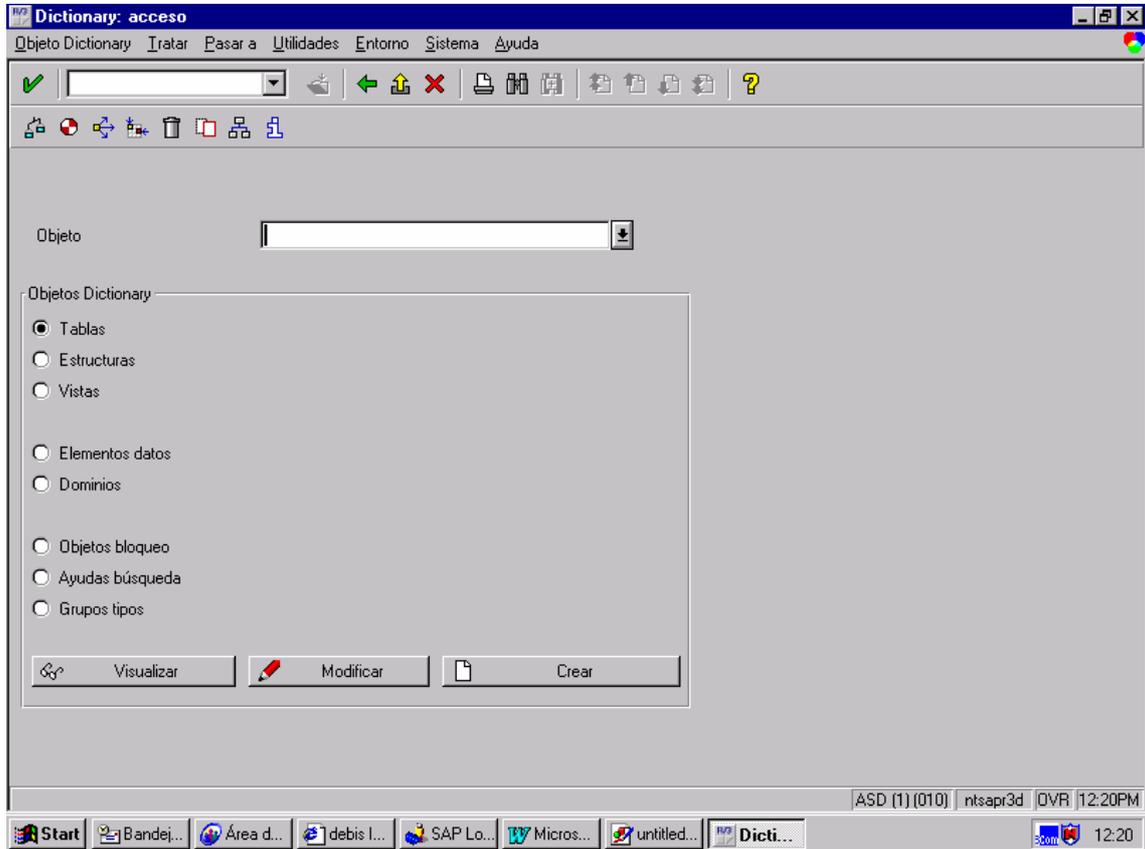
Browser Repository



TRANSACCIÓN → SE11

HERRAMIENTA → WORKBENCH ABAP →

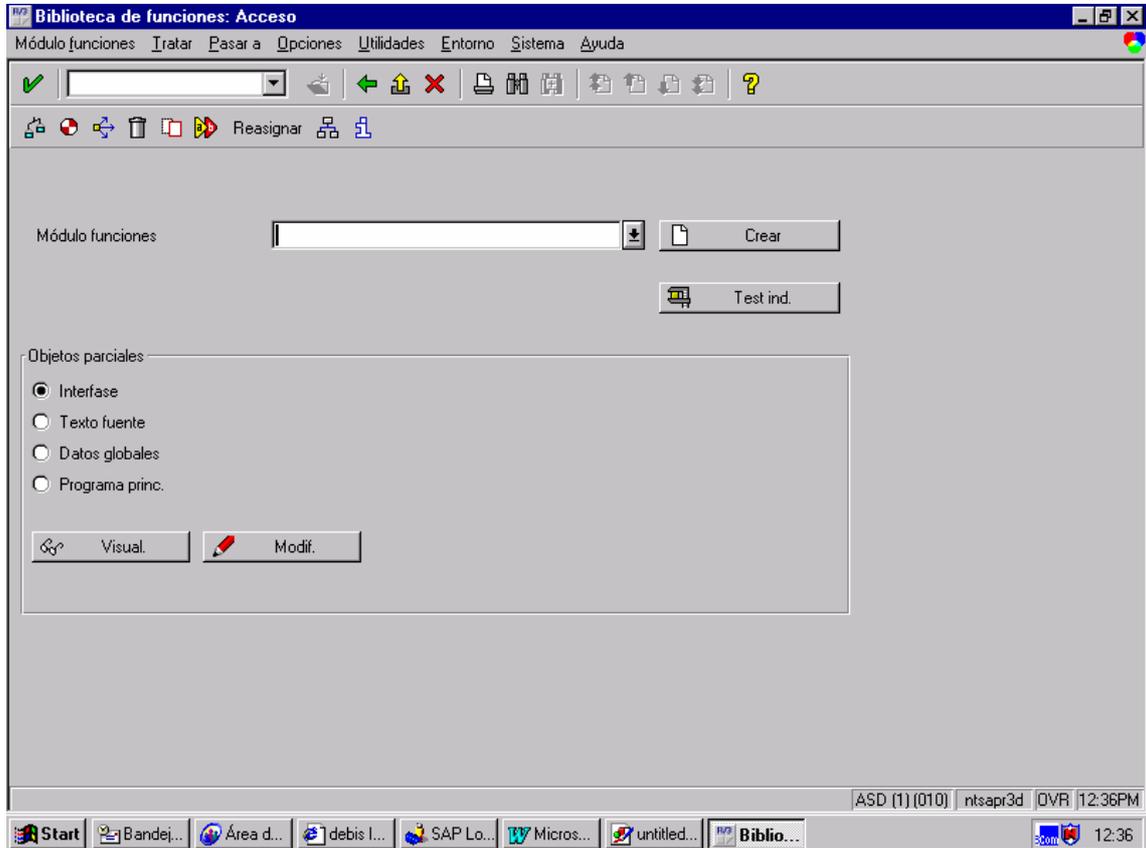
Dictionary ABAP



TRANSACCIÓN → SE37

HERRAMIENTA → WORKBENCH ABAP →

Biblioteca de funciones



PALABRAS CLAVE (Capítulo 5)

Keywords declarativas: Para declarar tipos de datos y objetos de datos (TYPES; DATA).

Keywords de evento: Para identificar el inicio de un bloque de proceso (conjunto de sentencias que se ejecutan cuando un suceso o evento ocurre (INITIALIZATION, START-OF-SELECTION).

Keywords de control: Identifican sentencias de control, con las que se pueden manejar el flujo de un programa (IF, WHILE).

Keywords operacionales: Identifican las sentencias de procesamiento de datos (MOVE, WRITE).

TIPOS DE DATOS (Capítulo 6)

Predefinidos Elementales: Forman parte del sistema.

C: Alfanumérico.

D: Fecha (AAAAMMDD).

F: Numérico en coma flotante.

I: Entero.

N: Texto no numérico.

P: Número empaquetado.

T: Hora (HHMMSS).

X: Hexadecimal.

Por el Usuario Elementales: Son creados a partir de los predefinidos elementales. Se utilizan con la sentencia TYPE.

Predefinidos Estructurados: Son las tablas definidas en el diccionario de datos.

Por el Usuario Estructurados:

Registros, 'field-strings': Agrupación de otros tipos de datos (TYPES).

Tablas Internas: Varias líneas del mismo tipo (OCCURS, TYPES).

OBJETOS DE DATOS

Internos: Creados en el programa y sin validez fuera de él (literales, estructuras y constantes).

Literales: Valor fijo. Literales de texto y numéricos.

Variables: Almacenan datos bajo un nombre y formato (DATA).

Constantes: Contiene un valor que no puede cambiar durante la ejecución del programa.

Externos: existen independientes del programa. Se almacenan sobre tablas de la base de datos del sistema SAP R7. (TABLES).

Definidos por el sistema: Constante SPACE y campos del sistema.

Especiales: Aparecen en la pantalla de selección.

Parámetros de selección (PARAMETERS)

Criterios de selección (SELECT-OPTIONS)

DECLARACION DE DATOS	
OBJETOS DE DATOS	TIPO DE DATOS
<p>TABLES Declaración de áreas de trabajo de tablas del diccionario de datos.</p> <p>DATA Declaración de variables, registros (field-strings), tablas internas y bloques de memoria común.</p> <p>CONSTAINS Declaración de variables y registros constantes.</p> <p>STATICS Declaración de objetos de datos estáticos a un procedimiento (subrutina o modelo de función).</p> <p>LOCAL Hace que el objeto de dato declarado anteriormente sea local dentro de una subrutina.</p> <p>FIELDS</p> <p style="padding-left: 20px;">FIELDS RANGES Permite definir tablas internas con la misma estructura que los criterios de selección o select-options.</p> <p style="padding-left: 20px;">FIELDS-GROUPS Declara extracto de datos, también llamado grupo de campos o field-groups. La agrupación de campos se define con la sentencia INSERT. INTO.</p> <p>DESCRIBE FIELD</p> <p>INSERT.INTO Define que campos formaran parte de un fiel-groups.</p> <p>FIELDS-SYMBOLS Declara un field-symbol, objeto de dato especial utilizado para la indirección. Es un puntero a un campo existente.</p> <p style="padding-left: 20px;">ASSIGN Asigna un objeto de dato a un field-symbol.</p> <p>CONTROLS Declaración de un control, objeto de dato especial.</p> <p>REFRESH CONTROL Inicia el objeto de dato especial de tipo de control.</p> <p>INFOTYPES Declara un Infotipo (RH)</p>	<p>TYPES Declaración de tipos de datos.</p> <p>TYPE-POOL Introduce la definición de un type-group.</p> <p>TYPE-POOLS Declara en un programa el type-group que vamos a utilizar.</p>
<p>PÁGINAS 76 Y SIGUIENTES</p>	

TABLAS INTERNAS (Capítulo 7)

Objetos de datos estructurados que permiten el almacenamiento de datos en memoria.

EXTRACTOS DE DATOS (Capítulo 7)

Objetos de datos estructurados que permiten el almacenamiento de datos en ficheros secuenciales en el área de almacenamiento del programa.

TABLAS INTERNAS Y EXTRACTOS	
TABLAS INTERNAS	EXTRACTOS
APPEND Añade registros al final de una tabla interna. Para copiar contenido de tablas internas.	INSERT INTO Libera el espacio reservado en memoria para la tabla interna.
INSERT Inserta nuevos registros antes de una línea existente en la tabla interna. Para copiar contenido de tablas internas.	EXTRACT Inserta registros en un extracto de datos.
MODIFY Modifica el contenido de un registro de una tabla interna.	LOOP END LOOP Lectura de registros de extractos de datos.
	AT END AT Eventos internos a la secuencia LOOP que se activan cuando se cumplen ciertas condiciones.
	SUM Suma los campos numéricos de los registros de un extracto de datos.
	CNT Calcula la media de los campos numéricos de los registros de un extracto de datos.
	SORT Ordena registros de un extracto de datos.
COLLECT Añade o modifica registros de una tabla interna en función de la clave estándar de la tabla.	
READ TABLE Lectura de un registro de una tabla interna utilizando un índice o una clave.	
LOOP AT.....END LOOP Lectura de registros de tablas internas, registro a registro.	
AT FIRST END AT se procesa en la primera iteración del bucle.	
AT LAST END AT se procesa en la última iteración del bucle.	
AT NEW END AT se procesa cuando se detecta un cambio de valor en el campo indicado.	
AT ENT END AT cuando toma por última vez un valor de la tabla.	
SUM Suma los campos numéricos de los registros de una tabla interna.	
CNT Calcula la media de los campos numéricos de los registros de una tabla interna.	
DELETE Borra registros de una tabla interna.	
PROVIDE END PROVIDE	
REFRESH Inicia el contenido de una tabla interna.	
DESCRIBE TABLE Recupera información de una tabla interna: número de registros, etc.	
SORT Ordena los registros de una tabla interna.	
MOVE Para copiar contenido de tablas internas, sobrescribiendo la tabla destino.	
WRITE TO Modifica el contenido de un registro de una tabla interna.	
CLEAR Limpia la línea de cabecera de la tabla interna	
FREE Libera el espacio reservado en memoria para la tabla interna.	



PÁGINAS 90 Y SIGUIENTES

FIELD-SYMBOLS (Capítulo 8)

Puntero a un campo existente. No tiene espacio físico reservado, ya que apunta a un campo que no es conocido hasta la ejecución del programa.

ASSIGN Asigna un objeto de dato a un field-symbol

PAGINAS 111 Y SIGUIENTES

EVENTOS (Capítulo 9)

Eventos Principales (Sistema):

INITIALIZATION Se procesa antes de que aparezca la pantalla de selección.
AT SELECTION-SCREEN Se procesa mientras la pantalla de selección está activa.
ON <parámetro>
ON END OF <criterio>
ON VALUE-REQUEST FOR <campo>
ON HELP-REQUEST FOR <campo>
ON RADIOBUTTON GROUP <botón>
ON BLOCK <bloque>
OUTPUT

START-OF-SELECTION Se procesa después de la pantalla de selección
GET Punto de control entre un report y una base de datos lógica. Posterior al evento anterior.

GET <tabla> [FIELDS <lista>]
GET <tabla> late
END-OF-SELECTION Se activa después de procesar todos los eventos GET.

Eventos activados en la fase de generación de la salida del report (listado):

TOP-OF-PAGE Se activa cuando el programa procesa la primera sentencia de escritura de cada página de un informe.

END-OF-PAGE Se activa cuando el programa procesa la última sentencia de escritura de cada página de un informe.

Eventos que se procesan después de la salida principal de un programa:

AT LINE-SELECTION Se activa cuando se selecciona una línea de la salida principal.
AT USER-COMMAND Se activa cuando se selecciona una función, o se introduce un comando en el campo de comandos.
AT pfnn Se activa cuando se presiona una tecla de función.

ABANDONO DE BLOQUES DE PROCESO

STOP Abandona cualquier bloque de proceso e ir directo al bloque de proceso del evento END-OF-SELECTION. Abandono de forma incondicional

EXIT Para terminar un bucle de forma incondicional.

CHECK Termina con la ejecución de un bloque de proceso de forma condicional y para finalizar el paso de un bucle de forma condicional.

CHECK <condición>

CHECK {<criterio-selección>|SELECT-OPTIONS}

REJECT Termina con el bloque de proceso del evento GET para ir a la siguiente iteración del mismo evento o de otro evento GET.

REJECT [<tabla>]

PAGINAS 121 Y SIGUIENTES
CAPITULO 30



EXPRESIONES LÓGICAS

OPERADORES PARA CUALQUIER OBJETO DE DATO

EQ	Igual a.
NE	No igual a.
LT	Menor que.
LE	Menor o igual que.
GT	Mayor que.
GE	Mayor o igual que.

OPERADORES PARA STRINGS CARÁCTER Y NUMERICOS

CO	Contiene sólo.
CN	No contiene sólo.
CA	Contiene algún.
NA	No contiene algún.
CS	Contiene el string.
NS	No contiene el string.
CP	Contiene el pattern. '*' cualquier número de caracteres. '+' cualquier carácter. '#'
NP	No contiene el pattern. '*' cualquier número de caracteres. '+' cualquier carácter. '#'

OPERADORES PARA ESTRUCTURAS DE BIT

O	Comprobación del valor 1. TRUE si para cada bit de valor '1', la posición respectiva vale '1'.
Z	Comprobación del valor 0. TRUE si para cada bit de valor '1', la posición respectiva vale '0'.
M	Comprobación de ambos valores. TRUE si para cada bit de valor '1', la posición respectiva vale '0' en alguna ocasión y '1' en alguna otra.

OPERADORES ESPECIALES

BETWEEN	AND	Rango de valores.
IS INITIAL		Valor inicial.
IN		Valor perteneciente a un criterio de selección.

OPERADORES LOGICOS

AND
OR
NOT

PAGINAS 133 Y SIGUIENTES



SENTENCIAS DE CONTROL (Capítulo 10).

SENTENCIAS DE CONTROL	
IF ENDIF	Bifurcación condicional.
ELSE	
ELSEIF	
CASE ENDCASE	Bifurcación múltiple condicional.
WHEN	
ON CHANGE ON ENDON	
DO ENDDO	Bucle incondicional.
WHILE ENDWHILE	Bucle condicional.
CHECK	Termina el paso de un bucle de forma condicional.
CONTINUE	Termina el paso de un bucle de forma incondicional.
EXIT	Termina un bucle incondicional.

SENTENCIAS DE PROCESAMIENTO NUMERICO

COMPUTE	Compute expresiones numéricas. La palabra clave es opcional.
ADD	Obtiene la suma de dos campos.
ADD-CORRESPONDING	Obtiene la suma de los componentes de un field-string.
SUBTRACT	Obtiene la diferencia de dos campos.
SUBTRACT-CORRESPONDING	Obtiene la diferencia de los componentes de un field-string.
MULTIPLY	Obtiene el producto de dos campos.
MULTIPLY-CORRESPONDING	Obtiene el producto de los componentes de un field-string.
DIVIDE	Obtiene la división de dos campos.
DIVIDE-CORRESPONDING	Obtiene la división de los componentes de un field-string.

OPERADORES

+	Suma	Resultado = campo1 + campo2.
-	Diferencia	Resultado = campo1 - campo2.
*	Multiplicación	Resultado = campo1 * campo2.
/	División	Resultado = campo1 / campo2.
DIV	División entera	Resultado = campo1 DIV campo2.
MOD	Resto de una división entera	Resultado = campo1 MOD campo2.
**	Exponenciación	Resultado = campo1 ** campo2.

FUNCIONES VALIDAS PARA TODOS LOS TIPOS NUMERICOS (F, I y P)
--

ABS	Valor absoluto de un argumento	Resultado = ABS(campo)
SIGN	Signo del argumento (1-positivo, 0-si es 0 y -1 si es negativo.	Resultado = SIGN(campo)
CEIL	Valor entero inferior al argumento	Resultado = CEIL(campo)
FLOOR	Valor entero superior al argumento.	Resultado = FLOOR(campo)
TRUNC	Parte entera del argumento	Resultado = TRUNC(campo)
FRAC	Parte fraccionaria del argumento	Resultado = FRAC(campo)

FUNCIONES VALIDAS PARA EL TIPO F

ACOS	Arcocoseno	Resultado = ACOS(campo)
ASIN	Arcoseno	Resultado = ASIN(campo)
ATAN	Arcotangente	Resultado = ATAN(campo)
COS	Coseno	Resultado = COS(campo)
SIN	Seno	Resultado = SIN(campo)
TAN	Tangente	Resultado = TAN(campo)
COSH	Coseno hiperbólico	Resultado = COSH(campo)
SINH	Seno hiperbólico	Resultado = SINH(campo)
TANH	Tangente hiperbólico	Resultado = TANH(campo)
EXP	Exponenciación	Resultado = EXP(campo)
LOG	Logaritmo natural (base e)	Resultado = LOG(campo)
LOG10	Logaritmo base 10	Resultado = LOG10(campo)
SQRT	Raíz cuadrada	Resultado = SQRT(campo)

SENTENCIAS DE MANEJO DE STRINGS

CONCATENATE	Concatena varios objetos de datos en un solo objeto de dato.
SPLIT	Divide el contenido de un objeto de dato en varios objetos de datos.
SHIFT	Permite realizar desplazamientos del contenido de un campo.
CONDENSE	Condensa el contenido de un objeto de dato.
TRANSLATE	Transforma el contenido de un objeto de dato.
TO UPPER CASE	
TO LOWER CASE	
USING	
OVERLAY	Sobreescribe el contenido de un objeto de dato sobre otro.
REPLACE	Reemplaza el contenido de un objeto de dato.
SEARCH	Realiza una búsqueda sobre un objeto de dato.

FUNCIONES VALIDAS PARA CAMPOS ALFANUMERICOS

STRLEN	Determina la longitud de un string hasta el último carácter distinto del espacio en blanco.	Resultado = STRLEN(campo)
--------	---	---------------------------

SALIDA ESTANDAR DE DATOS (Capítulo 12)

Sentencias para la generación de un listado.

SENTENCIAS BASICAS “sencillas”

WRITE	Sentencia básica de escritura.
ULINE	Subrayado.
SKIP	Salto de líneas.
NEW-LINE	Salto de líneas.
NEW-PAGE	Salto de página.
SET BLANK-LINES	Eliminación de la línea en blanco.

SENTENCIAS AVANZADAS “complicadas”

REPORT	Con esta sentencia se pueden definir ciertas características de un programa.
PROGRAM	Sentencia equivalente a la REPORT.
FORMAT	Da formato a la salida de un programa.
DETAIL	Salida en baja intensidad.
SUMMARY	Salida en alta intensidad.
RESERVE	Reserva líneas en la página actual de salida.
POSITION	Posiciona el puntero de escritura en una posición.
BACK	Retrocede el punto de inserción a una posición determinada.
SET LANGUAGE	Inicia todos los elementos de texto en el lenguaje especificado en la sentencia.
SET COUNTRY	Define el formato decimal (punto o coma para la separación de miles) y el formato de fecha para el país especificado.
SET MARGIN	Define un margen de impresión.
PRINT CONTROL	Escribe caracteres especiales de impresión,
SET LEFT SCROLL-BOUNDARY	Define las columnas que no se mueven en un desplazamiento horizontal.

PANTALLA DE SELECCIÓN (Capítulo 13)

Aparece al comienzo de la ejecución. Se introducen los parámetros y criterios de selección.

SENTENCIAS PARA COMPONER LA PANTALLA DE SELECCIÓN

PARAMETERS	Define un parámetro en la pantalla de selección.
SELECT-OPTIONS	Define un criterio de selección en la pantalla de selección.
SELECTION-SCREEN	Da formato a la pantalla de selección.

CAMPOS DE UNA TABLA DE SELECCION

SIGN	Flag. 'I'-Inclusión. 'E'-Exclusión
OPTION	Operador de selección. - EQ igual a. No se utiliza en el campo HIGH. - NE no igual a. No se utiliza en el campo HIGH. - GE mayor o igual que. No se utiliza en el campo HIGH. - GT mayor que. No se utiliza en el campo HIGH. - LE menor o igual que. No se utiliza en el campo HIGH. - LT menor que. No se utiliza en el campo HIGH. - CP contiene el patrón. Los caracteres especiales son '+' y '*'. No se utiliza en el campo HIGH. - NP no contiene el patrón. Los caracteres especiales son '+' y '*'. No se utiliza en el campo HIGH. - BT entre los valores. Se utiliza en los campos LOW y HIGH. - NB no entre valores. Se utiliza en los campos LOW y HIGH.
LOW	El tipo de dato LOW es el mismo que el campo referenciado en la sentencia SELECT-OPTIONS con la cláusula FOR. Si el campo HIGH está vacío, LOW indica valor sencillo. Si el campo HIGH tiene algún valor, LOW y HIGH marcan los límites del criterio de selección.
HIGH	El tipo de dato HIGH es el mismo que el campo referenciado en la sentencia SELECT-OPTIONS con la cláusula FOR. HIGH siempre se utiliza en combinación con el campo LOW.

SQL ABAP/4 (Capítulo 14)

“**Open SQL**”. Conjunto de sentencias propias del lenguaje ABAP/4 que encapsulan las sentencias SQL.

“**Native SQL**”. Sentencias nativas SQL incluidas en un programa ABAP/4.

ACCESO BASE DE DATOS	
OPEN SQL	NATIVE SQL
SELECT ENDSELECT. Lectura de datos de las tablas de la Base de Datos.	EXEC SQL ENDEXEC
INSERT. Añade líneas en tablas de la Base de Datos.	EXIT FROM SQL
UPDATE. Modifica líneas en tablas de la Base de Datos.	
MODIFY. Añade o modifica líneas en tablas de la Base de Datos.	
DELETE. Borra líneas en tablas de la Base de Datos.	
OPEN CURSOR. Abre un cursor sobre una tabla de la Base de Datos.	
FETCH. Avanza una posición un cursor de una tabla de la Base de Datos.	
CLOSE CURSOR. Cierra un cursor de una tabla de la Base de Datos.	
COMMIT WORK. Confirmación de modificaciones sobre la Base de Datos.	
ROLLBACK WORK. Deshace las modificaciones sobre la Base de Datos.	

PROGRAMAS INTERACTIVOS (Capítulo 15)

En un listado interactivo se pueden seleccionar líneas, realizar entradas, introducir comandos, presionar teclas de función, elegir opciones del menú o presionar botones de pulsación.

PROGRAMAS INTERACTIVOS	
SENTENCIAS	EVENTOS
MESSAGE	AT LINE SELECTION
SET PF-STATUS	AT USER-COMMAND
SET TITLEBAR	AT PFnn
SET USER-COMMAND	TOP-OF-PAGE DURING LINE-SELECTION
WINDOW	
HIDE	
GET CURSOR	
SET CURSOR	
DESCRIBE LIST	
SCROLL LIST	
READ LINE	
READ CURRENT LINE	
MODIFY LINE	
MODIFY CURRENT LINE	

CAMPOS DEL SISTEMA PARA LISTADOS INTERACTIVOS	
SY-LSIND	Índice del listado creado. Con el listado básico o principal la variable toma el valor "0", y se incrementa en 1 por cada listado secundario. Cuando abandonamos un listado secundario para ir al anterior listado la variable disminuye en 1.
SY-LISTI	Índice del nivel de listado desde el cual el evento fue activado.
SY-LILLI	Número absoluto de la línea desde la cual fue activado el evento.
SY-LISEL	Contenido de la línea desde la cual se activo el evento.
SY-CUROW	Posición de la línea en la ventana desde la cual se activó el evento (empieza a contar desde el valor 1).
SY-CUCOL	Posición de la columna en la ventana desde la cual el evento fue activado (empieza desde el valor 2).
SY-CPAGE	Número de página desde la cual se activa el evento.
SY-STARO	Número de la primera línea de la primera página mostrada del listado desde la cual fue activado el evento.
SY-STACO	Número de la primera columna mostrada en el listado desde la cual el evento fue activado.
SY-UCOMM	Código de función que activa el evento.
SY-PFKEY	Status del listado mostrado.

EVENTOS RELACIONADOS CON LISTADOS INTERACTIVOS	
AT LINE SELECTION	El evento se activa cuando el Usuario selecciona una línea.
AT USER-COMMAND	El evento se activa cuando el Usuario selecciona un comando.
AT PFnn	El evento se activa cuando el Usuario selecciona una tecla de función.
TOP-OF-PAGE DURING LINE-SELECTION	El evento se activa cuando el Sistema prepara los listados secundarios.

CODIGOS DE FUNCION PREDEFINIDOS

PFnn	Activa el evento PFnn.
PPn	Se pagina el listado hasta la página “n”.
PP-n	Se pagina el listado “n” páginas atrás.
PF+n	Se pagina el listado “n” páginas adelante.
PSn	Se desplaza el listado hasta la columna “n”.
PS--	Se desplaza el listado hasta la primera columna.
PS-n	Se desplaza el listado a la izquierda “n” columnas.
PS+n	Se desplaza el listado a la derecha “n” COLUMNAS.
PS++	Se desplaza el listado hasta la última columna.
PZn	Se pagina el listado hasta la línea “n”.
/	Se utiliza para comandos del sistema.

SENTENCIAS RELACIONADOS CON LISTADOS INTERACTIVOS

MESSAGE	Activa un mensaje.
SET PF-STATUS	Define el código de status, utilizado por el sistema para seleccionar el menú de pantalla asociado.
SET TITLEBAR	Define el título de pantalla que será incluido en el menú de pantalla en la primera línea.
SET USER-COMMAND	Activa un comando, por lo tanto, se activa el evento AT USER-COMMAND.
WINDOW	Define una ventana con formato de pantalla.
HIDE	Utilizado para esconder campos en líneas de salida en informes.
GET CURSOR	Recupera la posición del cursor.
SET CURSOR	Posiciona el cursor.
DESCRIBE LIST	Recupera información de un listado.
SCROLL LIST	Realiza un desplazamiento del listado.
READ LINE	Lee una línea de un listado.
READ CURRENT LINE	Lee la línea actual de un listado.
MODIFY LINE	Modifica el contenido de una línea.
MODIFY CURRENT LINE	Modifica el contenido de la línea actual.

MODULARIZACION DE UN PROGRAMA (Capítulo 16)

MODULARIZACIÓN DE UN PROGRAMA	
DECLARACION DE DATOS	
SENTENCIAS	SIGNIFICADO
INCLUDE STRUCTURE	Incluye una estructura del diccionario de datos.
INCLUDE TYPE	Incluye un tipo de dato.
TRANSACCION	
SENTENCIAS	SIGNIFICADO
CALL TRANSACCION	Llamada a una transacción sin abandonar el proceso actual.
LEAVE TO TRANSACTION	Llamada a una transacción abandonando el proceso actual.
PROGRAMA	
SENTENCIAS	SIGNIFICADO
INCLUDE	Incluye código.
SUBMIT	Ejecuta un programa.
LEAVE PROGRAM	Abandona el programa actual.
MODULOS DE DIALOGO	
SENTENCIAS	SIGNIFICADO
CALL DIALOG	Sentencia que llama a un módulo de dialogo.
MODULOS DE FUNCION	
SENTENCIAS	SIGNIFICADO
FUNCTION-POOL	Introduce un pool de funciones, también llamado grupo de funciones.
FUNCTION ENDFUNCTION	Define un módulo de función que será llamado por la sentencia CALL FUNCTION.
CALL FUNCTION	Realiza una llamada a un módulo de función.
CALL CUSTOMER-FUNCTION	Realiza una llamada a un módulo de función que ha sido definido por el Usuario y utilizado en un programa estándar.
RAISE	Activa una excepción. Sólo puede ser utilizada en el interior de un módulo de función.
LISTA	
SENTENCIAS	SIGNIFICADO
LEAVE TO LIST-PROCESSING	Cambia del modo de dialogo al modo de listado.
LEAVE LIST-PROCESSING	Abandona el modo de listado para retornar al modo de dialogo.
DYNPRO	
SENTENCIAS	SIGNIFICADO
CALL SCREEN	Llamada a un dynpro de la transacción.
SET SCREEN	Establece el siguiente dynpro que hay que presentar.
LEAVE SCREEN	Abandona el dynpro actual y lanza el dynpro que se haya establecido.
LEAVE TO SCREEN	Abandona el dynpro actual y lanza el dynpro indicado en la sentencia. Viene a ser una combinación de las sentencias SET SCREEN y LEAVE SCREEN.
LOOP AT SCEEN ENDLOOP	Bucle de lectura de los campos de pantalla.
MODIFY SCREEN	Modificación de los atributos de los campos de pantalla.
MODULO	
SENTENCIAS	SIGNIFICADO
MODULE ENDMODULE	Definición de un módulo.
SUPRESS DIALOG	Supresión de la pantalla de un dynpro. Desde el proceso PBO se pasa directamente al proceso PAI.
EXIT FROM STEP-LOOP	Salida incondicional de un bucle de pantalla (LOOP ENDLOOP en la lógica de proceso).
SUBRUTINA	
SENTENCIAS	SIGNIFICADO
FORM ENDFORM	Definición de una subrutina.



PERFORM

Llamada a una subrutina.

MACROINSTRUCCION	
SENTENCIAS	SIGNIFICADO

DEFINE END-OF-DEFINITION	Definición de una macroinstrucción. La ejecución de la macroinstrucción se realiza con el nombre de ésta.
-----------------------------	---

FUNCION EXTERNA	
SENTENCIAS	SIGNIFICADO

CALL	Sentencia que realiza una llamada a una función externa.
------	--

SALIDA DE PROCESOS	
SENTENCIAS	SIGNIFICADO

LEAVE	Abandona cualquier “modo interno” que haya sido generado por las sentencias CALL TRANSACTION, CALL DIALOG, SUBMID AND RETURN. El control se devuelve donde se realizo la llamada.
EXIT	Abandona una subrutina o cualquier unidad de modularización incondicionalmente.
CHECK	Abandona una subrutina o cualquier unidad de modularización condicionalmente.

DEPURACION DE UN PROGRAMA (Capítulo 17)

TRANSACCIONES PARA DEPURAR UN PROGRAMA	
SENTENCIAS	SIGNIFICADO
Transacción de depuración (debugging).	Permite ver el contenido de objetos de datos en tiempo de ejecución.
Transacción de análisis de tiempos de ejecución.	Permite analizar los tiempos de ejecución de un programa.
Transacción “Trace SQL”	Permite analizar las sentencias de acceso a la Base de Datos.
Transacción “log del sistema”.	Permite analizar los logs que se generan en el sistema.
Transacción de análisis de los “dumps”.	Permite analizar los “dumps” que se generan en el sistema.

SENTENCIAS DE DEPURACION DE UN PROGRAMA	
SENTENCIAS	SIGNIFICADO
BREAK	Con esta sentencia activamos la transacción de depuración de programas para un Usuario determinado.
BREAK-POINT	Sentencia utilizada para activar la transacción de depuración de programas.
GET RUN TIME	Sentencia utilizada para obtener el tiempo de ejecución de un programa.
SET RUN TIME	Sentencia que activa o desactiva la grabación de información en el fichero de análisis.
GET TIME	Con esta sentencia se obtiene la hora del sistema.
SET EXTENDED CHECK	Esta sentencia activa o desactiva la comprobación sintáctica de un programa.
SINTAX-TRACE	Esta sentencia activa o desactiva el trazador sintáctico.

INTERFAZ DE DATOS Y COMUNICACIONES (Capítulo 18)

SENTENCIAS DE ENTRADA/SALIDA EN MEMORIA SAP				
SENTENCIAS	SIGNIFICADO			
SET PARAMETER	Actualiza un parámetro de memoria.			
GET PARAMETER	Recupera un parámetro de memoria.			
SENTENCIAS DE ENTRADA/SALIDA EN MEMORIA ABAP/4				
Para traspasar información entre transacciones, entre programas o entre transacciones y programas.				
SENTENCIAS	SIGNIFICADO			
EXPORT TO MEMORY	Exporta un cluster de datos a memoria ABAP/4.			
IMPORT FROM MEMORY	Recupera un cluster de datos de la memoria ABAP/4.			
FREE MEMORY	Libera la memoria ABAP/4.			
SENTENCIAS DE ENTRADA/SALIDA EN MEMORIA INTERMEDIA DE APLICACIÓN				
SENTENCIAS	SIGNIFICADO			
EXPORT TO SHARED BUFFER				
IMPORT FROM SHARED BUFFER				
DELETE FROM SHARED BUFFER				
SENTENCIAS DE ENTRADA/SALIDA EN MEMORIA DE LA BASE DE DATOS				
SENTENCIAS	SIGNIFICADO			
EXPORT TO DATABASE	Exporta un cluster a un fichero de la Base de Datos.			
IMPORT FROM DATABASE	Importa un cluster de un fichero de la Base de Datos.			
DELETE FROM DATABASE	Borra un cluster de un fichero de la Base de Datos.			
IMPORT DIRECTORY FROM DATABASE	Recupera información de los clusters generados en los ficheros de la Base de Datos.			
Fichero Cluster Estándar				
NOMBRE	CL	TIPO	LG	DESCRIPCION
INDX-MANDT	X	CLNT	3	Mandante.
INDX-RELID	X	CHAR	2	Area.
INDX-SRTFD	X	CHAR	22	Clave del cluster.
INDX-SRTF2	X	INT4	4	Número identificativo de línea de cluster.
INDX-LOEKZ		CHAR	1	Marca de borrado (libre utilización).
INDX-SPERR		CHAR	1	Marca de bloqueo (libre utilización).
INDX-AEDAT		DATS	8	Fecha del último cambio.
INDX-USERA		CHAR	12	Usuario creador del cluster.
INDX-PGMID		CHAR	8	Programa creador del cluster.
INDX-BEGDT		DATS	8	Fecha de inicio (libre utilización).
INDX-ENDDT		DATS	8	Fecha final (libre utilización).
INDX-CLUSTR		INT2	2	Longitud de los datos.
INDX-CLUSTD		LRAW	2886	Datos comprimidos.
SENTENCIAS DE ENT/SAL EN FICHEROS EXTERNOS, EN EL SERVIDOR DE APLICACIÓN				
SENTENCIAS	SIGNIFICADO			
OPEN DATASET	Abre un Fichero Secuencial del servidor de aplicación.			
TRANSFER	Transfiere un registro al Fichero Secuencial del servidor de aplicación.			
READ DATASET	Lee un registro de un Fichero Secuencial del servidor de aplicación.			
DELETE DATASET	Borra un Fichero Secuencial del servidor de aplicación.			
CLOSE DATASET	Cierra un Fichero Secuencial del servidor de aplicación.			
EXPORT TO DATASET	Exporta datos a un Fichero Secuencial del servidor de aplicación. Esta sentencia no se debe utilizar.			
IMPORT FROM DATASET	Importa datos de un Fichero Secuencial del servidor de aplicación. Esta sentencia no se debe utilizar.			



SENTENCIAS ENT/SAL EN FICHEROS EXTERNOS, EN EL SERVIDOR DE PRESENTACION	
MODULO DE FUNCION	SIGNIFICADO
DOWNLOAD	Creación de un fichero externo sobre el servidor de presentación CON diálogo con el Usuario.
WS_DOWNLOAD	Creación de un fichero externo sobre el servidor de presentación SIN diálogo con el Usuario.
UPLOAD	Lectura de un Fichero externo del servidor de presentación CON diálogo con el Usuario.
WS_UPLOAD	Lectura de un Fichero externo del servidor de presentación SIN diálogo con el Usuario.
WS_QUERY	Obtiene información relacionada con los ficheros externos sobre el servidor de presentación.



UTILIZACION DE NOMBRES DE FICHEROS INDEPENDIENTES DEL SISTEMA

COMUNICACIÓN RFC

COMUNICACIÓN CPI-C

COMUNICACIÓN OLE2

LLAMADA A UN PROGRAMA EXTERNO

PROGRAMA DE BASE DE DATOS LOGICA (BDL) (Capítulo 19)

Una Base de Datos Lógica es un programa de acceso a la Base de Datos que se asocia a un programa principal. Los objetos parciales de una Base de Datos Lógica son:

ESTRUCTURA. Refleja las relaciones entre las tablas.

Estructura jerárquica.

Al nivel mas alto solo puede existir un nodo (nodo raíz).

Cada nodo puede tener una o más bifurcaciones.

Cada nodo solo puede derivar de un único nodo.

Para acceder a la pantalla con la estructura de la BDL → Transacción 'SE38' → Nombre del programa → Atributos → Visualizar → Aparece la BDL asociada a un programa.

SELECCIÓN. Determina un interfaz de Usuario asociado a la BDL.

PROGRAMA DE BASE DE DATOS. (SAPDBddd, ddd: Base de Datos).

Subrutinas, a las que el sistema llama en tiempo de ejecución del programa ABAP/4.

TEXTOS DE LA SELECCIÓN. Textos asociados a los campos de selección.

SELECCIÓN DE MATCHCODE. Objeto que permite asociar un "matchcode" a la BDL.

DOCUMENTACION. Objeto que permite asociar un texto de documentación a la BDL.

GENERACION DINAMICA DE OBJETOS (Capítulo 20).

GENERACION DINAMICA DE PROGRAMAS	
Manipular programas.	
SENTENCIAS	SIGNIFICADO
INSERT REPORT	Añade un programa en el sistema.
READ REPORT	Lee un programa del sistema.
DELETE REPORT	Borra un programa del sistema.
EDITOR-CALL FOR REPORT	
SYNTAX-CHECK	
GENERATE REPORT	Genera un programa del sistema.
GENERATE SUBROUTINE POOL	Genera un pool de subrutinas en memoria.
LOAD REPORT	
SCAN	
EXTENDED CHECK	

GENERACION DE ELEMENTOS DE TEXTO	
Textos independientes del código fuente del programa (Títulos de Programa, Cabeceras de Listado, Cabeceras de Columna, Textos de Selección, Textos Simbólicos).	
SENTENCIAS	SIGNIFICADO
INSERT TEXTPOOL	Inserta un elemento de texto en un programa del sistema.
READ TEXTPOOL	Lee un elemento de texto de un programa del sistema.
DELETE TEXTPOOL	Borra un elemento de texto de un programa.

GENERACION DE DYNPROS	
SENTENCIAS	SIGNIFICADO
EXPORT DYNPRO	Graba un dynpro en el sistema.
IMPORT DYNPRO	Lee un dynpro del sistema.
DELETE DYNPRO	Borra un dynpro del sistema.
SYNTAX-CHECK FOR DYNPRO	Verifica sintácticamente un dynpro.
GENERATE DYNPRO	Genera un dynpro.

GENERACION DE LOGS DE APLICACION (Capítulo 21).

Colector de mensajes. Necesario asociarlo a un objeto. Estructura del log:

CABECERA DE LOG. Número de log, Usuario, Fecha de generación, Programa o Transacción y Clase de mensaje.

MENSAJES DE LOG.

CREACION DE UN LOG DE APLICACION	
MODULO DE FUNCION	SIGNIFICADO
APPL_LOG_WRITE_HEADER	Escribe datos de la cabecera del log en la memoria local.
APPL_LOG_WRITE_LOG_PARAMETERS	Escribe el nombre de un parámetro de log y sus valores asociados en memoria local.
APPL_LOG_WRITE_MESSAGES	Escribe uno o más mensajes sin parámetros en memoria local.
APPL_LOG_WRITE_SINGLE_MESSAGE	Escribe un mensaje sin parámetros en memoria local.
APPL_LOG_WRITE_MESSAGE_PARAMS	Escribe un mensaje con parámetros en memoria local.
APPL_LOG_SET_OBJECT	Con este módulo de función creamos un nuevo objeto o subobjeto en memoria local. En función de un parámetro del módulo los mensajes serán escritos en la memoria local o en pantalla.
APPL_LOG_INIT	Comprueba si los objetos o subobjetos existen y, además, borra todos los datos asociados existentes de la memoria local.
APPL_LOG_INIT_MESSAGES	Borra los mensajes de la memoria local.
APPL_LOG_WRITE_DB	Escribe todos los datos asociados a un objeto de la memoria local a la Base de Datos.
LECTURA DE UN LOG DE APLICACION	
MODULO DE FUNCION	SIGNIFICADO
APPL_LOG_READ_DB	Lee los datos de un log de la Base de Datos para un objeto o subobjeto, según los parámetros especificados en el módulo de función.
APPL_LOG_READ_INTERN	Lee los datos de un log de la memoria local.
APPL_LOG_READ_INTERN_LONGTEXT	Prepara los textos estándar de las cabeceras de logs de aplicación.
BORRADO DE UN LOG DE APLICACION	
MODULO DE FUNCION	SIGNIFICADO
APPL_LOG_DELETE	Borra logs de la Base de Datos en función de los parámetros especificados en el módulo de función.
APPL_LOG_DELETE_WITH_LOGNUMBER	Borra logs de la Base de Datos en función del número de log.
VISUALIZACION DE UN LOG DE APLICACION	
MODULO DE FUNCION	SIGNIFICADO
APPL_LOG_DISPLAY	Permite analizar logs de aplicación de la Base de Datos.
APPL_LOG_DISPLAY_INTERN	Permite analizar logs de la aplicación de la memoria local.
AYUDAS DE LOG DE APLICACION	
MODULO DE FUNCION	SIGNIFICADO
APPL_LOG_PREPARE_HEADER_TEXT	Obtiene información de los datos de la cabecera de un log de aplicación.
APPL_LOG_PREPARE_MESSAGE_TEXT	Obtiene información de los textos de los mensajes de un log de aplicación.
MESSAGE_PREPARE	Obtiene información de los mensajes de un log de aplicación.



PROCESOS BATCH-INPUT (Capítulo 22)

Actualización de la Base de Datos a través de las transacciones del sistema de forma automática.

Ventajas:

- No requieren intervención manual durante la transferencia de datos.
- Integridad de datos.

PROCESOS BATCH-INPUT	
SENTENCIAS	MODULOS DE FUNCION
CALL TRANSACTION USING	BDC_OPEN_GROUP
CALL DIALOG USING	BDC_INSERT
	BDC_CLOSE_GROUP

Los métodos de batch-input se apoyan en tablas internas con la estructura BCDDATA

Estructura BCDDATA			
NOMBRE	TIPO	LG	DESCRIPCION
PROGRAM	CHAR	8	Nombre del modulpool de la transacción.
DYNPRO	NUMC	4	Nombre del dynpro de la transacción.
DYNBEGIN	CHAR	1	Inicio de datos de un dynpro.
FNAM	CHAR	35	Nombre de campo.
FVAL	CHAR	80	Valor de campo.

Identificación de dynpro. Registro inicial.

- PROGRAM: Nombre del modulpool de la transacción.
- DYNPRO: Número del dynpro.
- DYNBEGIN: "X".

Introducción de valores a un campo.

- FNAM: Nombre del campo
- FVAL: Valor del campo.

Ejecutar una función.

- Se introduce el código o el número de función (eje. /11, /15).
- El campo de comandos se llama BDC_OKCODE.
- Se utilizan los campos FNAM y FVAL.

Campos de un bucle.

- Se pone el índice del bucle entre paréntesis.
- Se utilizan los campos FNAM y FVAL.

Posicionamiento del cursor.

- FNAM: BDC_CURSOR
- FVAL: Nombre del campo donde situar el cursor.

METODO 'BATCH-INPUT CLASICO'.

Un programa ABAP/4 lee los datos externos que serán introducidos en el sistema y genera una sesión batch-input. En una sesión batch-input se almacenan las acciones que son requeridas para introducir datos a través de una transacción del sistema. Después de generar la sesión de batch-input hay que procesarla a través de la transacción 'SE56' (Sistema → Servicios → Batch-input → Tratar).

Se utilizan los siguientes módulos de función:

MODULO DE FUNCION	SIGNIFICADO
BDC_OPEN_GROUP	Se crea una sesión de batch-input.
BDC_INSERT	Para insertar transacciones en la sesión de batch-input. La tabla interna con estructura BDCDATA contendrá los datos necesarios para actualizar la información con una transacción.
BDC_CLOSE_GROUP	Para cerrar una sesión de batch-input.

Cuando un programa crea un juego de datos de batch-input, éste permanece en la cola de batch-input hasta su procesamiento. Los juegos de datos se procesan de dos formas:

TRANSACCION DE GESTION DE BATCH-INPUTS

A través de la transacción 'SE56' (Sistema → Servicios → Batch-input → Tratar)

PROGRAMA RSBDCSUB

METODO 'CALL TRANSACTION'.

No genera juegos de datos de batch-input, las transacciones son ejecutadas on-line.

Procesa un juego de datos compuesto de una sola transacción directamente en un programa, sin utilizar la transacción de gestión de juegos de datos. Es más rápido que el clásico pero con el inconveniente del seguimiento de errores.

Se utiliza la sentencia CALL TRANSACTION USING.

METODO 'CALL DIALOG'.

Se basa en el uso de los módulos de dialogo.

Se utiliza la sentencia CALL DIALOG USING.

No se debe utilizar.

PROCESOS 'BACKGROUND' (Capítulo 23).

- Gestionar jobs (mostrar, copiar o borrar jobs).
- Comprobar o accionar eventos (Ej. Arrancar jobs).
- Mostrar los logs generados por un job.

METODO BASICO	
Simplifican la generación y la gestión de los procesos de fondo.	
MODULO DE FUNCION	SIGNIFICADO
BP_JOBVARIANT_SCHEDULE	Se utiliza para planificar un job para su ejecución. 'Método express'.
BP_JOBVARIANT_OVERVIEW	Ofrece una interfaz simplificada para la gestión de jobs.

METODO MANUAL	
La creación se realiza paso a paso.	
MODULO DE FUNCION	SIGNIFICADO
JOB_OPEN	Se utiliza para crear un job o proceso background (de fondo).
JOB_SUBMIT	Se utiliza para añadir pasos a un job que haya sido abierto por el módulo de función JOB_OPEN.
SUBMIT	Se utiliza para ejecutar un programa.
JOB_CLOSE	Se utiliza para pasar el job al sistema de proceso de fondo para ser ejecutado.
GET_PRINT_PARAMETERS	Se utiliza para indicar los parámetros de impresión de un job.
BP_CALCULATE_NEXT_JOB_STARTS	Se utiliza para calcular la fecha y hora en la cual un job planificado podría ser ejecutado.
BP_JOB_COPY	Se utiliza para copiar procesos de fondo.
BP_JOB_MAINTENANCE	Llama a la transacción de resumen de jobs 'SE38'.
BP_JOB_SELECT	Permite seleccionar jobs.
BP_FIND_JOBS_WITH_PROGRAM	Obtiene aquellos jobs que utilizan un programa determinado.
BP_JOBLIST_PROCESSOR	Se utiliza para presentar al Usuario una lista de jobs para su mantenimiento.
BP_JOBLOG_READ	Se utiliza para leer el log de un job.
BP_JOBLOG_SHOW	Presenta información sobre los logs existentes en el sistema.
BP_START_DATE_EDITOR	Se especifica cuando y cómo un job será arrancado.
SHOW_JOBSTATE	Obtiene el estado de un job.

EVENTOS	
Arrancan procesos de fondo. Existen eventos del Sistema (lanzados por el sistema) y definidos por el Usuario.	
MODULO DE FUNCION	SIGNIFICADO
BP_CHECK_EVENTID	Se utiliza para comprobar si el evento especificado ha sido definido en el sistema (código de retorno a 0).
BP_EVENT_RAISE	Acciona un evento en el sistema de proceso de fondo.
SAPEVT	Activa un evento externamente al sistema SAP. Puede ser utilizado desde un script externo, un fichero BATCH o un programa.





CONTROL DE AUTORIZACIONES (Capítulo 24).

CONTROL DE AUTORIZACIONES	
VALIDACION	SIGNIFICADO
AUTHORITY CHECK	Permite comprobar las autorizaciones de un Usuario.

Las Bases de Datos Lógicas estándar validan las autorizaciones de los Usuarios.

Las sentencias OPEN SQL o SQL nativas no activan ningún tipo de validación de autorización.

ESTRUCTURA DE UNA TRANSACCION Capitulo 25).

ELEMENTOS.

- Código de Transacción.**
- Dynpro.** (dynamic program)
 - Pantalla.** Se definen los campos (E, S, E/S)
 - Lógica de proceso.** Programa
 - Menú.**
- Modulpool.** Programa contenedor de módulos.

SECCIONES DE UNA PANTALLA

- Sección Menú.**
 - Título.**
 - Barra de Menús.**
 - Barra de funciones estándar.**
 - Barra de funciones adicionales.**
- Sección Screen.**
- Barra de Estado.**

TRANSACCIONES NECESARIAS PARA CREAR UNA TRANSACCION

- Mantenimiento de transacciones.** Transacción SE93. Diferentes tipos.
 - Transacción de diálogo.** Transacción típica ABAP/4.
 - Transacción report.** Se asocia a la transacción un report.
 - Transacción variantes.** Se asocia una transacción con una variante de transacción.
 - Menú de área.** Se crea un menú de ámbito (donde solo esta activa la barra de menús).
 - Transacción parámetros.** Esta transacción permite realizar llamadas a otras transacciones pasando parámetros a la pantalla inicial.

Mantenimiento de dynpros. Transacción 'sreen painter'.

Un dynpro está compuesto:

- Pantalla.**
- Lógica de Proceso.**
 - PBO (Process Before Output).** Procesos que toman parte antes de mostrar la pantalla.
 - PAI (Process After Input).** Procesos que intervienen después de mostrar la pantalla.

Mantenimiento de menús. Transacción 'menú painter'. Se mantienen los menús de pantalla.

Mantenimiento de modulpools.

Selección de campos.

SENTENCIAS DE LA LOGICA DE PROCESO	
SENTENCIA	SIGNIFICADO
PROCESS BEFORE OUTPUT	Define el bloque de proceso PBO.
PROCESS AFTER INPUT	Define el bloque de proceso PAI.
PROCESS ON HELP-REQUEST	Para sustituir el proceso estándar de ayuda sobre un campo.
PROCESS ON VALUE-REQUEST	Para sustituir el proceso estándar de valores posibles sobre un campo.
MODULE	Para llamar a un módulo del programa 'modulpool'.
FIELD	Para validar campos de la pantalla.
CHAIN	Para agrupar campos de pantalla. Si se activa un mensaje dentro de un bloque CHAIN ENDCHAIN, todos los campos definidos en ese bloque aparecen de entrada (modificables).



CALL SUBSCREEN
LOOP

Para realizar llamadas a una subscreen.
Para realizar lecturas de bucles de pantalla.



SENTENCIAS DEL MODULPOOL	
SENTENCIA	SIGNIFICADO
MODULE ENDMODULE	Definición de un módulo.
MESSAGE	Activa un mensaje del sistema.
LOOP AT SCREEN	Realiza un bucle de lectura de los campos de pantalla.
MODIFY SCREEN	Modifica los atributos de un campo de pantalla.
SUPP'RESS DIALOG	Suprime la aparición de la pantalla. Une la ejecución del bloque PBO y PAI sin que el sistema presente la pantalla del dynpro. Esta sentencia solo tiene sentido en el bloque PBO.

MANEJO DE ERRORES (Capítulo 26).

TECNICAS DE CONTROL DE ERRORES.

Comprobaciones automáticas del sistema.

Entrada automática. Cuando un campo se define como obligatorio.

Formato de campo. El sistema valida los campos en función de cómo esté definido.

Valores válidos para un campo. A través del diccionario de datos.

Limitación de valores.

Existencia en la tabla de referencia de un campo que es clave foránea.

Sentencia FIELD. Permite validar las entradas realizadas sobre un campo.

Cláusula VALUE. Validación lógica de proceso.

Cláusula MODULE. Validación en el modulpool.

Sentencia MESSAGE. Se activa un mensaje. Diferentes tipos:

Tipo E (error). Si no se soluciona el problema no se puede pasar al dynpro siguiente.

Tipo W (warning). El proceso puede continuar tras un 'INTRO'.

Tipo I (information). Tras pulsar 'INTRO' se queda el dynpro. Mensaje informativo

Tipo A (abend). Error grave del sistema que provoca el abandono de la transacción.

Tipo S (success). Presenta el mensaje en la pantalla siguiente.

CONEXION TRANSACCION-PROGRAMA (Capitulo 27).

TRANSACCION → MODO DIALOGO
PROGRAMA → MODO LISTADO

OBJETO	CREA UN NUEVO NODO	NO CREA UN NUEVO NODO
TRANSACCION	CALL TRANSACTION Cuando la transacción llamada termine devolverá el control a la transacción o programa que realiza la llamada.	LEAVE TO TRANSACTION La transacción o programa que realiza la llamada termina su ejecución.
PROGRAMA	SUBMIT AND RETURN Cuando el programa termine devolverá el control a la transacción o programa que realiza la llamada.	SUBMIT La transacción o programa que realiza la llamada termina su ejecución.

CONEXIÓN ENTRE EL MODO DIALOGO Y EL MODO LISTADO.

Dos métodos:

Creando un modo interno con la sentencia SUBMIT y la cláusula AND RETURN.
Sin abandonar la transacción actual con la sentencia LEAVE TO LIST-PROCESSING.
Para retornar del modo listado al modo diálogo se utiliza la sentencia LEAVE LIST-PROCESSING.

AYUDAS DE PANTALLA (Capítulo 28).

Ayudas orientadas a conocer la funcionalidad de un campo de pantalla (F1 o 'botón de ?').

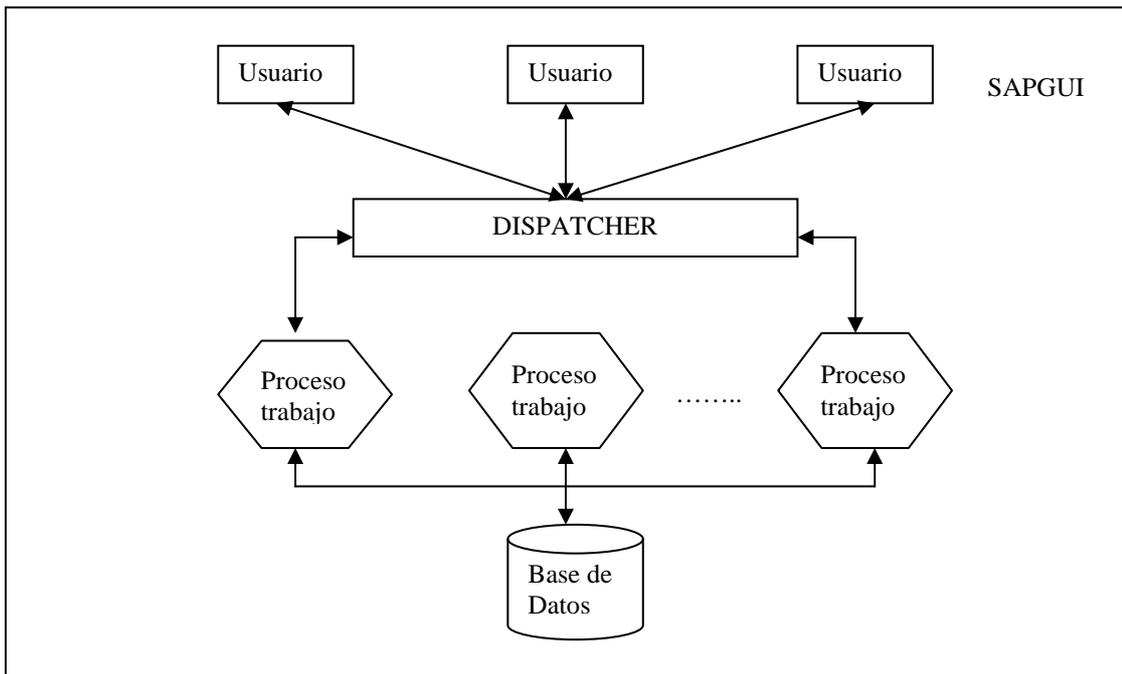
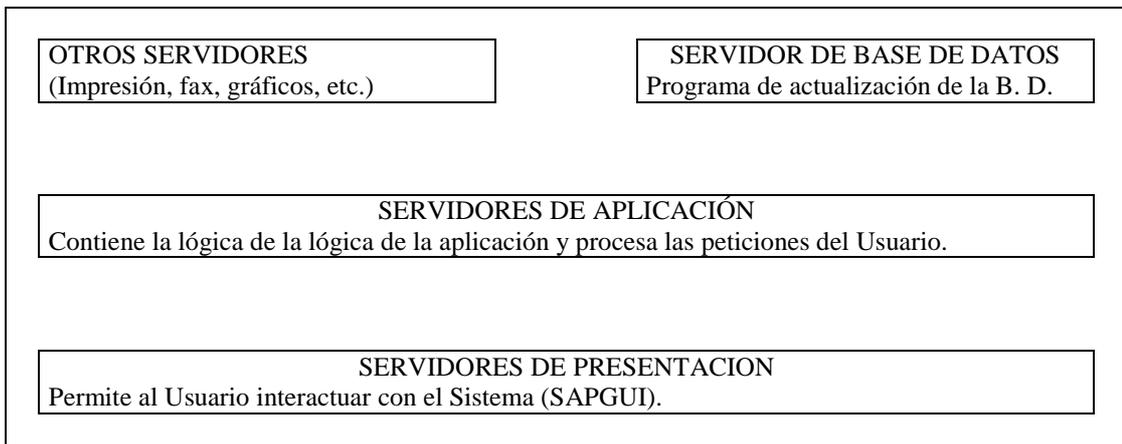
Ayudas para introducir valores sobre un campo (F4 o 'entradas posibles').

AYUDAS DE PANTALLA	
MODULO DE FUNCION	SIGNIFICADO
POPUP_TO_CONFIRM_LOSS_OF_DATA	
POPUP_TO_CONFIRM_STEP	
POPUP_TO_CONFIRM_WITH_MESSAGE	
POPUP_TO_CONFIRM_WITH_VALUE	
POPUP_TO_DECIDE	
POPUP_TO_DECIDE_WITH_MESSAGE	
POPUP_TO_DISPLAY_TEXT	
POPUP_TO_DECIDE_LIST	
POPUP_NO_LIST	
SAPGUI_PROGRESS_INDICATOR	

ACTUALIZACIONES DE LA BASE DE DATOS (Capítulo 29).

ARQUITECTURA CLIENTE/SERVIDOR

Tres niveles:



DISPATCHER. Se encarga de recoger las peticiones de los Usuarios y distribuirlas entre los Procesos de Trabajo

PROCESOS DE TRABAJO. Programa que se encarga de procesar las tareas de la Aplicación (peticiones de los Usuarios). Diferentes procesos:

- Procesos de Trabajo de Diálogo.
- Procesos de Trabajo de Batch.
- Procesos de Trabajo de Spool.



Procesos de Trabajo de Gestión de Bloqueos.
Procesos de Trabajo de Actualización.

TRANSACCION. Acción que permite al Usuario realizar una modificación en la Base de Datos.

- Transacción de Base de Datos. LUW (Logical Unit Work).
- Transacción de Actualización. También denominada SAP LUW.
- Transacción ABAP/4. Conjunto de SAP LUWs agrupadas en una sola transacción ABAP/4

MECANISMOS DE ACTUALIZACION

TECNICAS ESPECIALES DE ACTUALIZACION	
SENTENCIA	SIGNIFICADO
PERFORM ON COMMIT	La rutina especificada en la sentencia PERFORM se ejecuta en la misma tarea de diálogo cuando el sistema ejecuta la sentencia COMMIT WORK
CALL FUNCTION IN UPDATE TASK	El módulo de función se ejecuta en un proceso de actualización cuando el sistema ejecuta la sentencia COMMIT WORK.
CALL FUNCTION IN BACKGROUND TASK COMMIT WORK	El módulo de función se ejecuta en un proceso de fondo cuando el sistema ejecuta la sentencia COMMIT WORK. Activa las rutinas o módulos de función que hayan sido 'previstos' para ser ejecutados con esta sentencia. Con la cláusula AND WAIT el proceso espera a que se ejecuten todos los procesos de actualización que se encuentran en espera (actualización síncrona), Sin la cláusula el proceso de diálogo continua (actualización asíncrona). La actualización síncrona da mas seguridad, ya que se puede comprobar si la actualización se ha realizado correctamente.
ROLLBACK WORK	Anula cualquier rutina o módulo de función que esté 'preaviso' para ser ejecutado cuando cuando se ejecute la sentencia COMMIT WORK.
SET UPDATE TASK LOCAL	Se activa la actualización en una tarea local.

BLOQUEOS LOGICOS

Proceso de diálogo. Los datos son introducidos o modificados por el Usuario.
 Proceso de actualización. Se modifica la Base de Datos.

La operación asíncrona entre estos dos procesos obliga a bloquear los datos.

BLOQUES LOGICOS	
MODULO DE FUNCION	SIGNIFICADO
ENQUEUE_<objeto-bloqueo>	Módulo de función para bloquear el <objeto-bloqueo>.
DENQUEUE_<objeto-bloqueo>	Módulo de función para desbloquear el <objeto-bloqueo>.