

## Utilidades MVS (II)

### Entradilla:

Este mes se continúa con el tema dedicado a las Utilidades que acompañan al S.O. MVS de IBM. Estas utilidades son herramientas que por su sencillez y potencia permiten realizar gran número de funciones a nivel del sistema. Son las típicas utilidades del sistema operativo.

### 1.- Introducción

El pasado mes se inicio un tema de mucha utilidad en una gran instalación, tal y como puede comprobar cualquier desarrollador de este entorno a diario. Las utilidades que se trataron en ese primer artículo, fueron:

**IEFBR14** : Para inicializar o borrar ficheros.

**IEBGENER** : Para copiar ficheros y manipular su contenido.

**IEHLIST** : Para listar nombres de ficheros o miembros.

Estas utilidades, el autor del artículo se ha permitido clasificarlas como *utilidades de la primera época*, ya que acompañan al sistema operativo, desde sus primeras versiones, incluso, con su predecesor, el DOS/VS.

No es intención del autor trasladar a este artículo el manual de referencia de estas utilidades, pero la experiencia demuestra que la barrera que existe en estas instalaciones entre el entorno de sistemas y el entorno de desarrollo, impide el acceso físico a dichos manuales, lo cual hace que se den casos tan paradójicos como el que se le presentó días antes de escribir este artículo, cuando cayó en sus manos un JCL que 'para borrar un fichero' invocaba a NATURAL, y una vez en esta entorno, haciendo uso de las facilidades del producto NATURAL-PROCESS borraba dicho fichero, con lo cual lo que se esta haciendo es elevar el consumo de recursos a límites insospechados por tener que *alocar*, al menos, todos los ficheros del entorno NATURAL, y todo ello por no conocer la utilidad IEFBR14 vista el pasado mes.

Particularmente se recomienda insistir, en todas aquellas instalaciones en las que no se disponga de los manuales de IBM, para conseguir una copia de los mismos, ya que copiar las distintas utilidades desde otro JCL que haga algo parecido no resuelve el problema, pues obligará probablemente a dar mas pasos de los necesarios; o bien, dependiendo de la iniciativa del programador, a utilizar los medios del entorno que este conozca, como en el ejemplo comentado anteriormente, aunque esto sea mas costoso en recursos del sistema.

### 2.- Reglas de las fichas SYSIN

Los comandos que debe ejecutar la utilidad se agrupan en la Ddname SYSIN, la cual suele ir In-stream (es decir, en la propia cadena), aunque puede perfectamente especificarse dentro de cualquier otro fichero, asociando su Dsname a esta Ddname.

A modo de recordatorio, se citan las reglas que deben seguirse para especificar estos comandos:

- El **nombre del comando** tiene que ser un nombre valido para la utilidad.
- Los operandos van separados del comando por al menos un blanco.
- Los distintos **operandos** de un comando, van concatenados entre sí, sirviendo la coma (,) como delimitador de los mismos.
- El tercer grupo de elementos de una ficha de control son los **comentarios**, que se encuentran separados de los operandos por al menos un blanco.
- La codificación de las fichas de control no puede sobrepasar la columna 71 (Ojo con la numeración propia del fichero que contiene el JCL, que puede provocar un error en el caso de que tenga definido NUM STD ON)
- Cuando no cabe la codificación de una ficha en una línea, se detiene la codificación en la columna 71, o después de una coma, y se escribe algo

(distinto de blanco) en la columna 72, continuando esta codificación en la siguiente línea a partir de la columna 16.

### 3.- Caracteres 'ASA'

El control de carro para la impresión de ficheros se realiza incorporando estos caracteres ASA en la primera columna de aquellos ficheros que tengan definido en la DCB (Description Control Block) el formato de registro (Parámetro RECFM) FBA o VBA (**F**ijo/**V**ariable **B**loquado **A**SA)

Los caracteres ASA con significado son los que se muestran en la **figura 1**.

<b>B</b>	Imprime la línea después de un salto de carro
<b>O</b>	Imprime la línea después de dos saltos de carro
<b>-</b>	Se imprime la línea después de tres saltos de carro
<b>N</b>	Se imprime la línea sin saltar el carro (sob reimprime)
<b>+N</b>	Se sob reimprime la línea N veces sin saltar el carro.
<b>1</b>	Se salta de Página.
<b>*</b>	La línea no se imprime por tratarse como comentario.

**Figura 1 : Caracteres ASA**

### 4.- IEHPROG

Esta utilidad permite:

- Borrar, catalogar, descatalogar y renombrar un fichero; y
- Borrar y renombrar un miembro de un fichero particionado.

Los comandos admitidos por esta utilidad a través de la **SYSIN** son:

- **CATLG** :Cataloga un fichero.
- **UNCATLG** :Descataloga un fichero.
- **SCRATCH** :Borrar un fichero.
- **RENAME** :Renombra un fichero.

Como ya se ha repetido en varias ocasiones, un fichero se encuentra catalogado cuando su nombre figura en un fichero de la instalación que contiene todos los nombres de los ficheros de la instalación y, para cada uno de ellos, el nombre del volumen (o disco físico) en el que se encuentra físicamente. Por lo tanto, para catalogar un fichero, solo hay que especificar dos parámetros, DSN (Data\_Set\_Name) o nombre del fichero, y el nombre del volumen en el que se encuentra.

Los siguientes ejemplos muestran como catalogar dos ficheros, uno en disco (SYSDA) y otro en cinta (TAPE):

**CATLG DSN=JMPDES.FICHEROD,VOL=SYSDA=VOLUM11**

**CATLG DSN=JMPDES.FICHEROC,VOL=TAPE=(002365,1,000153,1)**

Por el contrario, para descatalogar un fichero, solo hay que borrarle del catalogo, por lo tanto, solo hay que especificar el nombre del fichero a descatalogar. Es de notar que solo se borra la entrada del catalogo, pero no se borra el fichero, el cual seguirá residiendo en el disco o cinta en el que se encuentre. Ejemplo:

**UNCATLG DSN=JMPDES.FICHEROD**

Para borrar un fichero físicamente, se debe utilizar el comando **SCRATCH**. Este comando accede directamente a la VTOC (Volume Table Of Contains - Tabla de contenido del Volumen) del volumen que se especifica como parámetro, y borra la entrada correspondiente, liberando el espacio usado. Ejemplo:

**SCRATCH DSN=JMPDES.FICHEROD,VOL=SYSDA=VOLUM11**

Así mismo, se recuerda que los ficheros particionados están formados por dos partes, el área de directorio, que contienen los nombres o índice de los miembros, y el área de datos, que contiene los datos, o contenido propio de los distintos miembros. Por lo tanto, cuando se desea borrar un miembro de un PDS (Particioned Data Set) se debe especificar el nombre del fichero y el volumen en el que se encuentra para poder localizar el área de directorio en la VTOC. Y una vez localizado, borrar de dicho área la entrada correspondiente al miembro a borrar, pero no se libera el espacio hasta que se comprima la librería (Solo Programadores num. 19). Ejemplo:

**SCRATCH DSN=JMPDES.FICHEROP,VOL=SYSDA=VOLUM11,MEMBER=EJEMPLO1**

Por ultimo, para renombrar ficheros, se usa el comando RENAME, y los parámetros DSN para especificar el nombre actual, y NEWNAME para especificar el nombre nuevo.

Cuando se trata de usar este comando para renombrar miembros de un PDS, se debe especificar el nombre del miembro y el parámetro NEWNAME.

**RENAME DSN=JMPDES.FICHERO1,VOL=SYSDA=VOLUM11,NEWNAME=JMPDES.FICHERO9**

**RENAME DSN=JMPDES.FICHEHROP,VOL=SYSDA=VOLUM11,  
MEMBER=EJEMPLO1,NEWNAME=EJEMPLO9**

Esta utilidad también tiene como particularidad la citada el pasado mes con la utilidad **IEHLIST**, en el sentido de que este programa para poder ejecutarse, debe contener fichas DD para alocar (incorporar al entorno) el Volumen que se referenciará en el comando a ejecutar en la SYSIN, tal y como muestra el ejemplo de la figura 2.

```
//EJEMPLO1 EXEC PGM=IEHPROG
//SYSPRINT DD *
CATLG      DSN=JMPDES.FICHERO1,VOL=SYSDA=VOLUM11
UNCATLG    DSN=JMPDES.FICHERO2
SCRATCH    DSN=JMPDES.FICHERO3,VOL=SYSDA=VOLUM11,PURGE
RENAME     DSN=JMPDES.FICHERO4,VOL=SYSDA=VOLUM11,
           MEMBER=VIEJO,NEWNAME=NUEVO
/*
//VOLUMEN  DD UNIT=SYSDA,VOL=SER=VOLUM11,DISP=SHR
```

**Figura 2 : Ejemplo de la utilidad IEHPROG**

## 5.- IEBPTPCH

Este programa de utilidad se usa para imprimir o perforar en ficha o cinta de teletipo:

- Un fichero secuencial o particionado completo.
- Miembros específicos de un fichero particionado
- Registros seleccionados de un fichero secuencial o particionado.
- El directorio de un fichero particionado
- Versión editada de un fichero secuencial o particionado.

El nombre del fichero de entrada se especifica asociándole a la Ddname SY-SUT1, y el de salida a la SYSUT2, que normalmente coincidirá con una cola de salida del spool, la cual, por defecto tiene la siguiente DCB: DCB=(LRECL=121,BLKSIZE=121,RECFM=FBA).

Los comandos admitidos por esta utilidad a través de la Ddname **SYISIN** son:

- **PRINT** Imprime un fichero.
- **RECORD** Especifica el formato del registro de salida.
- **MEMBER** Especifica el nombre del miembro a listar.
- **TITLE** especifica el literal de la cabeceras de las paginas.

Los operadores del comando **PRINT** son:

**TYPORG=PO** para Organización Particionada y **PS** para organización Secuencial  
**MAXFLDS=n** donde **n** especifica el numero máximo de parámetros **FIELD** que se especificaran con el comando **RECORD**.

**MAXNAME=m** donde **m** especifica el numero máximo de parámetros **NAME** que se especificaran con el comando **MEMBER**

**STOPAFT=nn** donde **nn** especifica el numero de registros que se listaran de cada miembro.

**SKIP=ss** donde **ss** especifica que imprima una línea en blanco cada **ss** registros impresos.

**STRTAFT=pp** donde **pp** especifica que empiece a imprimir ignorando los **pp** registros iniciales.

Los operadores del resto de comandos se muestran junto con un ejemplo:

a) Imprimir los miembros que se especifican:

**MEMBER NAME=miembro1,NAME=miembro2**

b) Imprimir los primeros 80 bytes del registro de entrada.

**RECORD FIELD=(80)**

c) Imprimir 5 asteriscos en las primeras posiciones, y en las posiciones 6 y siguientes los 10 bytes que se encuentran en las posiciones 71 y siguientes del registro de entrada.

**RECORD FIELD=(5,'\*\*\*\*\*',,1),FIELD=(10,71,,6)**

d) Imprimir un literal como titulo dejando un offset de 40 posiciones, es decir, dejando 40 blancos a la izquierda:

**TITLE ITEM=('CABECERA LISTADO',40)**

```

/*=====
/* IMPRIMIR FICHERO CON LOS NOMBRES DE LOS OBJETOS A LISTAR
/*=====
//CABECERA EXEC PGM=IEBTPCH
//STEPLIB DD DSN=SYS1.LINKLIB,DISP=SHR
//SYSPRINT DD SYSOUT=*
/* ----- FICHERO DE ENTRADA -----
//SYSUT1 DD *
*****XXXXXXXX
    LA ENTRADA ESTA FORMADA POR ESTAS LINEAS DE
    TEXTO Y POR EL FICHERO JMPDES.ENTRADA
*****XXXXXXXX
/*
//          DD DISP=SHR,DSN=JMPDES.ENTRADA
/* ----- FICHERO DE SALIDA -----
//SYSUT2 DD SYSOUT=*,HOLD=YES
//SYSIN DD *
    PRINT TYPORG=PS,MAXFLDS=1
    RECORD FIELD=(40,1,,30)
    TITLE ITEM=('DSN : JMPDES.ENTRADA',30)
    TITLE ITEM=('*****',30)
/*

```

Figura 3 : Ejemplo de IEBTPCH

La **figura 3** muestra un ejemplo de uso de esta utilidad, mediante la cual se desea poner un titulo al listado del fichero que contiene los nombres de los objetos a listar.

## 6.- IEBCOPY

Esta utility es usada para:

- copiar ficheros particionados a secuencial,
- Restaurar ficheros desde cinta a disco.

Si la DD de entrada esta asociada a un fichero particionado, y la DD de salida esta asociada a un fichero secuencial copia todo el fichero PDS sobre el secuencial de salida.

La particularidad de esta utilidad se centra en el hecho de que los nombres de los ficheros de entrada y de salida no se referencian asociándoles a las ddnames SYSUT1 o SYSUT2, sino que se referencian especificando ddnames en las fichas DD que contienen los nombres de estos ficheros, y luego referenciando estas ddnames en el comando, como se vera a continuación.

Los comandos admitidos por esta utilidad a través de la **SYSIN** son:

- **COPY** :Define las Ddnames asociadas a los ficheros de entrada (**INDD**) y de salida (**OUTDD**). El siguiente ejemplo concatena dos librerías de entrada.

**COPY INDD=(DDENTRA,DDENTRA2),OUTDD=DDSALIDA**

- **SELECT** :Define los nombres de los miembros a copiar mediante el parámetro **MEMBER=(miembro1,miembro2,...)**

- **EXCLUDE** :Especifica que se copien todos los miembros de la librería de entrada, excluyendo los miembros cuyos nombres se relacionan con el parámetro **MEMBER**.

**EXCLUDE MEMBER=(miembro4,miembro5)**

Cuando en la librería de salida existen los miembros especificados con la ficha **SELECT**, no se copian los ficheros. No obstante, puede especificarse que reemplace al existente mediante el subparametro **R**, tal y como muestra el siguiente ejemplo:

**SELECT MEMBER=(MIEMBRO1,(miembro2,,R))**

Cuando se desea que un miembro de la librería de entrada se copie con otro nombre sobre la librería de salida, se usara el segundo subparametro.

```
SELECT MEMBER=( (miembro2,nom_nuevo,R),miembro3)
```

#### **7.- IEBUPDTE**

Esta utilidad se usa para la creación y actualización de ficheros secuenciales o particionados a partir de registros lógicos de 80 caracteres.

Las operaciones posibles son:

- Incorporación de un nuevo miembro en un PDS.
- Sustitución de un miembro ya existente.
- Adición de un nuevo registro a un miembro o secuencial
- Borrado de un registro lógico
- Conversión de un fichero secuencial en un particionado.

Esta utilidad esta algo desfasada, razón por la que no se van a describir sus comandos, y solo se han descrito sus funciones, por si algún lector tuviera necesidad de hacer uso de ella, saber qué utilidad debe emplear.

#### **8.- Próxima articulo:**

Puesto que el espacio no perdona, se aplaza para el próximo mes el resto de programas con un gran peso especifico en cualquier gran instalación, los programas **IDCAMS** e **IKJEFT01** o programa TSO para ejecutar en batch.

## 9.- Utilidad

La utilidad que acompaña este mes al artículo es algo especial ya que hace uso de la utilidad **IEBGENER** para submitir un JOB.

La utilidad de esta utilidad, valga la redundancia, radica en el hecho de saltarse los controles que establecen las instalaciones cuando se hace uso del comando **SUBMIT** de TSO, ya que escribe directamente en la cola interna de entrada de trabajos (INTRDR: Internal Reader).

### Clist para editar un JCL, cambiarle y submitir resultado

```
PROC      2 FICHERO NODO
SET &TEMPFIL = &SYSUID..JMP.TMP
IF &SYSDSN('&TEMPFIL') = OK THEN DELETE '&TEMPFIL'
EDIT 'JMPDES.JCL.PROTOTIP(IEBXXVAX)' OLD NONUM CNTL
TOP
C * 999 'FFFFFF' '&FICHERO' ALL
TOP
C * 999 'NNNNNN' '&NODO' ALL
SAVE '&TEMPFIL'
END NOSAVE
ALLOC FI(SYSUT1) DA('&TEMPFIL') SHR REUSE
ALLOC FI(SYSUT2) SYSOUT(A) WRITER(INTRDR) REUSE
ALLOC FI(SYSIN) DUMMY REUSE
ALLOC FI(SYSPRINT) DA(*) REUSE
CALL 'SYS1.LINKLIB(IEBGENER)'
FREE FILE(SYSUT1 SYSUT2 SYSPRINT SYSIN)
DELETE '&TEMPFIL'
WRITE *****
WRITE ===>> FICHERO : &FICHERO ENVIADO A &NODO < OK >
WRITE *****
ACAABA: EXIT
END
```

Figura U1

La **figura U1** muestra el listado de una CLIST o procedimiento de comandos TSO, que pide 2 datos: Fichero y nodo. A continuación:

- edita internamente el fichero o prototipo de JCL mostrado en la **figura U2**, y realiza las sustituciones de FFFFFFF por el valor dado a la variable fichero, y NNNNNN por el valor dado a la variable Nodo.
- salva el resultado de la edición sobre un fichero auxiliar
- aloca el fichero temporal dándole como ddname SYSUT1
- aloca como salida la cola "A" de entrada de trabajos, asignándola la Ddname SYSUT2
- invoca al programa IEBGENER que copia SYSUT1 sobre SYSUT2
- borra el fichero temporal
- devuelve un mensaje de fichero enviado.

Esta CLIST suponiendo que se encontrara contenida en el fichero secuencial 'JMPDES.ENVIAR' ejecutando desde la línea de comandos:

**TSO EX 'JMPDES.ENVIAR'**

**JCL usado como prototipo en la clist de la figura U1.  
Este JCL tiene por DSN=JMPDES.JCL.PROTOTIP[IBMXXVAX]**

```
//JMPDESIE JOB (12,345), 'ENTORNO DESARROLLO', CLASS=T, MSGCLASS=X,
//          NOTIFY=JMPDES, MSGLEVEL=(0,0)
//*****
//SUBMITIR EXEC PGM=IEBGENER, COND=(0,LT)
//SYSUT2 DD SYSOUT=(A,INTRDR)
//SYSPRINT DD SYSOUT=*
//SYSIN DD DUMMY
//SYSUT1 DD DATA, DLM=WB
//JMPPROIE JOB (34,567), 'ENTORNO PRODUCCION', CLASS=K, MSGCLASS=X,
//          NOTIFY=JMPDES, MSGLEVEL=(0,0), RESTART=*
//** ***** **
//** ***** SE TRANSMITE EL FICHERO A DIGITAL ***** ITLIMITE **
//** ***** **
//EJECUTAR EXEC PGM=IKJEFT01
//SYSEXEC DD DSN=..., DISP=SHR
//SYSPRT DD SYSOUT=*
//SYSTSIN DD *
... FICHAS DE CONTROL DE LA UTILIDAD A EJECUTAR
SEND FFFFFFF NNNNNN
/*
WB
```

**Figura U2**

En el disco que acompaña a la revista también se incluye una referencia al libro, inédito hasta la fecha, escrito por el autor de este artículo, titulado 'TSO PARA DESARROLLADORES/JCL/CLIST/ISPF'