

Traspaso de ficheros
FORTRAN de datos
de VAX/VMS a
ALPHA/UNIX

E. Sánchez,
B. Ph. van Milligen

Toda correspondencia en relación con este trabajo debe dirigirse al Servicio de Información y Documentación, Centro de Investigaciones Energéticas, Medioambientales y Tecnológicas, Ciudad Universitaria, 28040-MADRID, ESPAÑA.

Las solicitudes de ejemplares deben dirigirse a este mismo Servicio.

Los descriptores se han seleccionado del Thesaurus del DOE para describir las materias que contiene este informe con vistas a su recuperación. La catalogación se ha hecho utilizando el documento DOE/TIC-4602 (Rev. 1) Descriptive Cataloguing On-Line, y la clasificación de acuerdo con el documento DOE/TIC.4584-R7 Subject Categories and Scope publicados por el Office of Scientific and Technical Information del Departamento de Energía de los Estados Unidos.

Se autoriza la reproducción de los resúmenes analíticos que aparecen en esta publicación.

Depósito Legal: M-14226-1995

NIPO: 238-97-001-5

ISSN: 1135-9420

Editorial CIEMAT

CLASIFICACIÓN DOE Y DESCRIPTORES

990200

DATA TRANSMISSION, DATA PROCESSING, FORTRAN, DEC COMPUTERS, CAMAC
SYSTEM

"Traspaso de ficheros FORTRAN de datos de VAX/VMS a ALPHA/UNIX"

Sánchez, E.; van Milligen, B.Ph.

23 pp. 1 figs. 5 refs.

Resumen

Se han desarrollado varias herramientas software con objeto de acceder a la base de datos de TJ-I y TJ-IU, residente en servidores VAX/VMS, desde el servidor DEC ALPHA 8400 del Sistema de Adquisición de Datos del TJ-II.

Las bases de datos de TJI/TJ-IU son heterogéneas y contienen varios tipos de ficheros, SADE, CAMAC y FORTRAN sin formato. Las herramientas presentadas aquí permiten transferir ficheros CAMAC y ficheros FORTRAN sin formato, de acuerdo a los tipos normalizados definidos en este documento, desde un servidor VAX/VMS para la explotación de tales datos en el servidor ALPHA/Digital UNIX.

"FORTRAN datafiles transference from VAX/VMS to ALPHA/UNIX"

Sánchez, E.; van Milligen, B.Ph.

23 pp. 1 figs. 5 refs.

Abstract

Several tools have been developed to access the TJ-I and TJ-IU databases, which currently reside in VAX/VMS servers, from the TJ-II Data Acquisition System DEC ALPHA 8400 server.

The TJ-I/TJ-IU databases are not homogeneous and contain several types of datafiles, namely, SADE, CAMAC and FORTRAN unformatted files. The tools presented in this report allow one to transfer CAMAC and those FORTRAN unformatted files defined herein, from a VAX/VMS server, for data manipulation on the ALPHA/Digital UNIX server.

Traspaso de ficheros FORTRAN de datos de VAX/VMS a ALPHA/UNIX

E. Sánchez , B. Ph. van Milligen

Resumen

Se han desarrollado varias herramientas software con objeto de acceder a la base de datos de TJ-I y TJ-IU, residente en servidores VAX/VMS, desde el servidor DEC ALPHA 8400 del Sistema de Adquisición de Datos del TJ-II.

Las bases de datos de TJI/TJ-IU son heterogéneas y contienen varios tipos de ficheros, SADE, CAMAC y FORTRAN sin formato. Las herramientas presentadas aquí permiten transferir ficheros CAMAC y ficheros FORTRAN sin formato, de acuerdo a los tipos normalizados definidos en este documento, desde un servidor VAX/VMS para la explotación de tales datos en el servidor ALPHA/Digital UNIX.

FORTRAN datafiles transference from VAX/VMS to ALPHA/UNIX

E. Sánchez , B. Ph. van Milligen

Abstract

Several tools have been developed to access the TJ-I and TJ-IU databases, which currently reside in VAX/VMS servers, from the TJ-II Data Acquisition System DEC ALPHA 8400 server

The TJ-I/TJ-IU databases are not homogeneous and contain several types of datafiles, namely, SADE, CAMAC and FORTRAN unformatted files. The tools presented in this report allow one to transfer CAMAC and those FORTRAN unformatted files defined herein, from a VAX/VMS server, for data manipulation on the ALPHA/Digital UNIX server.

PALABRAS CLAVE: FORTRAN, DATA TRANSMISSION, DATA

INDICE

Introducción	5
Traspaso de ficheros de datos desde VAX/VMS a ALPHA/Digital UNIX	6
Traspaso manual de ficheros CAMAC/VMS	8
La librería libcamac	9
Traspaso manual de ficheros DATU, DATRU y CATA	10
Funcionamiento de getvax y mgetvax	10
Forma de uso de los procedimientos getvax y mgetvax	12
Transferencia automática de ficheros CAMAC, DATU, DATRU y CATA desde código FORTRAN	14
Apéndice A. Notas referentes a los formatos de los datos	15
Apéndice B: Tipos DATU, DATRU y CATA	16
El tipo DATU	16
El tipo DATRU	17
El tipo CATA	18
Referencias	19



Introducción

El Sistema de Adquisición de Datos del TJ-II debe proporcionar herramientas para el manejo y análisis de los datos tomados por los diferentes sistemas de diagnóstico.

En dicho sistema los datos se almacenan en un servidor central modelo DEC ALPHA 8400 con sistema operativo Digital UNIX. Este servidor, además de gestionar los datos tomados en el TJ-II, proporciona librerías matemáticas y gráficas para el análisis y visualización de datos.

Hasta el momento, gran cantidad de datos se encontraban almacenados en servidores VAX con sistema operativo VMS, tanto la base de datos de TJ-I o TJ-IU, como datos tomados en máquinas de otros laboratorios con los que la Unidad de Fusión tiene colaboración. Sería, por tanto, útil para los usuarios del servidor ALPHA 8400 poder traspasar de forma sencilla ficheros de datos desde los servidores VAX/VMS al servidor 8400 con el fin de aprovechar las capacidades de proceso de este equipo (muy superiores a las de los equipos VAX), en el análisis de datos antiguos.

En los servidores VAX/VMS existe gran variedad en cuanto al formato y organización de los ficheros de datos. Existen ficheros ASCII, ficheros FORTRAN sin formato, ficheros con organización de acceso secuencial, ficheros indexados, etc.

Las bases de datos de TJI /TJ-IU están compuestas por dos tipos de ficheros, ficheros CAMAC[1] , que son de acceso secuencial y ficheros SADE, con ficheros secuenciales, relativos e indexados. También se dispone de librerías de rutinas FORTRAN para el acceso a esos datos desde código FORTRAN: la librería SADE y la librería CAMAC.

Aparte de las bases de datos de TJI/TJ-IU existen gran cantidad de ficheros **FORTRAN sin formato** que contienen datos de máquinas de otros laboratorios (ATF, W7-AS, PBX, JET...), datos procesados, etc..

Con el fin de facilitar el acceso a esos datos desde el servidor ALPHA 8400, se han desarrollado unas herramientas que permiten transferir algunos de esos tipos de ficheros de datos de forma relativamente transparente y cómoda para el usuario.

Las herramientas presentadas aquí permiten transferir **ficheros FORTRAN sin formato** de acuerdo a los tipos normalizados definidos en el Apéndice B y ficheros **CAMAC**.

El presente informe pretende ser un resumen de las herramientas desarrolladas. Asimismo, puede servir como manual para el uso de esas herramientas.

Traspaso de ficheros de datos desde VAX/VMS a ALPHA/Digital UNIX

El traspaso de ficheros de datos entre sistemas VAX/VMS y ALPHA/Digital UNIX, no es siempre inmediato y las tareas a realizar en cada caso dependen del tipo de fichero de datos de que se trate.

Los ficheros ASCII son transferibles entre estos sistemas y se pueden leer sin mayor problema, otra situación es la que se presenta con los ficheros que contienen datos no formateados (binarios).

La representación binaria de los datos difiere entre VAX/VMS y ALPHA/digital UNIX, por lo que datos binarios transferidos vía FTP, por ejemplo, entre estas plataformas no son directamente incorporables a un código FORTRAN.

El DEC FORTRAN v3.0 (instalado en el servidor ALPHA 8400) incluye extensiones que permiten la lectura y conversión de datos binarios transferidos desde un VAX para su incorporación a un código corriendo sobre arquitectura ALPHA[2,3]¹. No obstante, estas extensiones sólo resuelven el problema de interpretar los bits de un número binario escrito en VMS, para ser cargado en la memoria de un ALPHA/Digital UNIX. Para ficheros de datos binarios además se necesita conocer su estructura interna, registros, número y tipo de datos que contienen, para poder interpretar la información correctamente.

Los ficheros FORTRAN de datos se dividen en registros, cuyo tamaño puede ser prefijado por el usuario, o no, al abrir el fichero desde código FORTRAN.

Para el caso de ficheros con longitud de registro fija, conocido el número de datos que contiene cada registro, y su tipo, una vez transferido el fichero de forma binaria, se pueden leer desde código FORTRAN haciendo uso de las extensiones que el DEC FORTRAN v3.0 para Digital UNIX proporciona. Este es el caso de los ficheros CAMAC de la base de datos de TJ-I/TJ-IU.

¹ Existen tres formas de leer datos escritos en VMS:

- usar la variable de entorno FORT_CONVERTn
- la opción de compilación -vms
- usar la opción CONVERT en la apertura del fichero

(vease DEC FORTRAN, Lenguaje Reference Manual y DEC FORTRAN, User Manual for DEC OSF/1 ALPHA Systems)

En el caso de ficheros en los que no se especifica la longitud de registro en la apertura del fichero, ficheros **DATU**, **DATRU** y **CATA** (ver Apéndice B) por ejemplo, se añade un registro en el fichero cada vez que se hace una escritura, registro que puede ser de “longitud variable” o “segmentado”, dependiendo de la cantidad de datos que contenga. Dado que la longitud de estos registros no es siempre la misma, se necesita añadirles cabeceras y pies para identificar su principio y su fin. Estas cabeceras y pies son introducidas por el sistema de ficheros, y son transparentes para el usuario.

Las cabeceras y pies introducidos en registros de longitud variable son diferentes en el caso de VMS y Digital UNIX, por lo que ficheros que contengan este tipo de registros necesitan de conversiones, además de la conversión de los datos binarios, para su traspaso desde VMS a UNIX.

En VMS, un registro de tipo “longitud variable” se compone de una cabecera de dos octetos y un área de datos de tamaño variable; en la cabecera está codificado el tamaño en octetos del área de datos². Un registro de tipo “segmentado”, es un registro sin límite de tamaño, en principio, que contiene uno o varios registros de tipo “longitud variable” (segmentos); cada uno de estos segmentos va encabezado por dos octetos que contienen información del lugar que ocupa el segmento dentro del registro (primer segmento, último, intermedio o único)[4,5].

La escritura de registros de esta clase desde código FORTRAN en Digital UNIX difiere en que las cabeceras de registro no contienen dos octetos, sino cuatro, y en que los registros de longitud variable contienen, además de una cabecera de 4 octetos un pie con otros 4 octetos[3,4]. Si se transfiere (vía FTP por ejemplo) un fichero de este tipo desde un servidor VAX/VMS a un ALPHA/Digital UNIX y se intenta leer sin previa modificación, se producirán errores, ya que el Digital UNIX espera encontrar cuatro octetos de cabecera y estos ficheros contienen sólo dos. Se precisa pues una conversión del fichero antes de intentar leerlo.

Para el traspaso de ficheros de datos CAMAC, DATU, DATRU y CATA, se han desarrollado herramientas que se encargan de transferir de forma automática el fichero desde el servidor VAX/VMS, y de realizar las conversiones necesarias en cuanto a registros, para que los datos del fichero puedan ser cargados a la memoria del servidor ALPHA, y procesados/analizados.

² En VMS, por defecto, el tamaño máximo de un registro de longitud variable es de 2044 octetos.

Traspaso manual de ficheros CAMAC/VMS

Los ficheros CAMAC de la base de datos de TJ-I y TJ-IU son ficheros FORTRAN de acceso secuencial [1], con una longitud de registro fija; esto permite que una vez transferidos en forma binaria desde el servidor VAX/VMS al ALPHA/Digital UNIX, se puedan leer desde código FORTRAN haciendo uso de las extensiones que proporciona el DEC FORTRAN para Digital UNIX [2,3].

Para el acceso a los datos almacenados en este tipo de ficheros de la base de datos de TJ-I/TJ-IU se desarrolló en su día una librería de rutinas FORTRAN, la cual residía en un servidor VAX/VMS.

Se ha creado un procedimiento, al nivel del sistema operativo, (**getcam**) para transferir de forma automática ficheros de este tipo desde un servidor VAX/VMS, y por otra parte se han adaptado las rutinas FORTRAN de la librería CAMAC/VMS necesarias para la lectura de estos ficheros, integrándolas en una nueva librería pública (**libcamac**) en el servidor ALPHA.

El funcionamiento de este procedimiento se describe a continuación:

getcam establece una conexión FTP automática con el servidor VAX/VMS haciendo uso del fichero **.netrc**³ de UNIX y transfiere en forma binaria el fichero requerido.

La sinopsis para **getcam** es la siguiente:

getcam *discharge* [*device*]

discharge es el número de descarga cuyos datos se quieren transferir.

device es una cadena de tres caracteres que identifica la máquina en la que se tomaron los datos (TJ1 para TJI o TJU para TJ-IU). Este parámetro es opcional y si no se usa, por defecto, se entenderá que es TJU.

Los datos se buscan siempre en el servidor VMS **dec**⁴, que es el que contiene los datos de TJ-I/TJ-IU, por lo que en el fichero **.netrc** debe aparecer una línea con este nombre de máquina, un nombre de usuario y una clave en el servidor (CIEVX1)⁵.

³ Ver información sobre FTP y **.netrc** en el manual de UNIX

⁴ Se utiliza este nombre para identificar el servidor (dec.ciemat.es) de dirección IP 130.206.11.6, que es el que contiene las bases de datos de TJI y TJ-IU.

La librería libcamac

Una vez transferido un fichero con **getcam** la lectura de los datos se puede hacer por medio de las rutinas **camac_signal** y **construct_fname**, para lo que se necesita enlazar el programa con la librería **libcamac**, ubicada en el directorio **/usr/local/lib** del servidor DEC 8400.

Además de las rutinas de lectura de datos, se ha instalado la utilidad **camac_info**, que permite obtener información de la descarga.

Para cualquier información sobre la librería CAMAC, o la estructura de los ficheros CAMAC, ver referencia 1.

Además de **camac_signal** y **construct_fname** se ha incluido una rutina llamada **getvaxfile** cuyo funcionamiento y parámetros de entrada se describe más adelante.

⁵ Puesto que el fichero **.netrc** contiene el nombre de usuario y la clave de una cuenta en otra máquina, es conveniente que este fichero no tenga permisos de lectura para otros usuarios con el fin de preservar la privacidad de estos datos.

Traspaso manual de ficheros DATU, DATRU y CATA

El lenguaje de programación FORTRAN permite el uso, sobre cualquier sistema operativo, de ficheros sin formato.

Estos ficheros presentan la ventaja frente a los ficheros con formato de que ocupan mucho menos espacio, razón por la que su utilización es muy común.

A pesar de que el estándar FORTRAN permite su manejo sobre cualquier sistema operativo, estos ficheros no son transferibles de forma inmediata, entre diferentes sistemas operativos y/o plataformas, como se expuso anteriormente.

Estos ficheros, cuando no se ha especificado un tamaño de registro en su apertura, internamente están organizados en registros segmentados o de longitud variable, cuyo número y tamaño depende de como se escribieron los datos en el fichero.

Se han definido varios “tipos normalizados” de ficheros FORTRAN sin formato, **DATU**, **DATRU** y **CATA** (ver Apéndice B), todos ellos útiles para el almacenamiento de trazas temporales en las que se necesita conocer el valor instantáneo de una magnitud con un periodo de muestreo fijo. Estos ficheros están compuestos de dos registros, uno de longitud variable y otro segmentado. La estructura de los ficheros de este tipo se detalla en el Apéndice B, en el que se incluyen además ejemplos de escritura y lectura de ficheros de acuerdo a estos tres tipos.

Para hacer la transferencia y conversión de estos ficheros de forma transparente para el usuario, se han desarrollado dos procedimientos, **getvax** y **mgetvax**, a nivel del sistema operativo UNIX, que se encargan de transferir desde el servidor VAX/VMS los ficheros (vía FTP) y hacer la “conversión a ALPHA/Digital UNIX”.

Estos procedimientos son susceptibles de ser llamados desde una rutina FORTRAN por medio de una llamada al sistema⁶, o bien haciendo uso de las **getvaxfile**.

Funcionamiento de **getvax** y **mgetvax**

Los procedimientos **getvax** y **mgetvax** se encargan de varias tareas:

1) Establecen una conexión FTP automática desde el servidor ALPHA con el VAX para transferir en forma binaria el(los) fichero(s) seleccionado(s).

⁶ Una vez transferido un fichero FORTRAN sin formato y “convertido” con **getvax** o **mgetvax** para leerlo desde código FORTRAN en ALPHA/UNIX se necesita utilizar las extensiones al efecto proporcionadas por el DEC FORTRAN para ALPHA/UNIX.(vease DEC FORTRAN, Lenguaje Reference Manual y DEC FORTRAN, User Manual for DEC OSF/1 ALPHA Systems).

2) Una vez transferido un fichero de uno de los tipos definidos arriba, se sustituyen las cabeceras de registro que introduce VMS por las cabeceras y pies correspondientes a Digital UNIX, lo que se hace del siguiente modo:

2.1. Se cuenta el número total de octetos que contiene el fichero, incluídas las cabeceras de registro de VMS.

2.2. Se añaden al fichero una cabecera y un pie de 4 octetos, conteniendo el número de octetos del fichero original. Con esto se obtiene un fichero FORTRAN sin formato con un solo registro de "longitud variable al estilo de UNIX"⁷ accesible desde código FORTRAN en ALPHA/Digital UNIX.

2.3. Desde un código FORTRAN, se abre el fichero y se leen los datos octeto por octeto, saltando las cabeceras de registro de VMS.

2.4. Se escriben desde código FORTRAN dos registros, uno con los 12 primeros octetos leídos, (registro de longitud variable), y otro con el resto de octetos de datos, (registro segmentado).

El uso de código FORTRAN para la escritura del fichero final, garantiza que el fichero podrá después ser leído desde código FORTRAN.

La figura 1 presenta esquemáticamente del proceso de conversión llevado a cabo por **getvax** y **mgetvax**.

mgetvax es similar a **getvax**, con la diferencia de que permite hacer el trabajo con varios ficheros simultáneamente.

⁷ En Digital UNIX un registro de longitud variable puede tener un tamaño máximo de $2^{31}-1$ octetos, suficiente para almacenar todo el fichero original en un registro de este tipo.

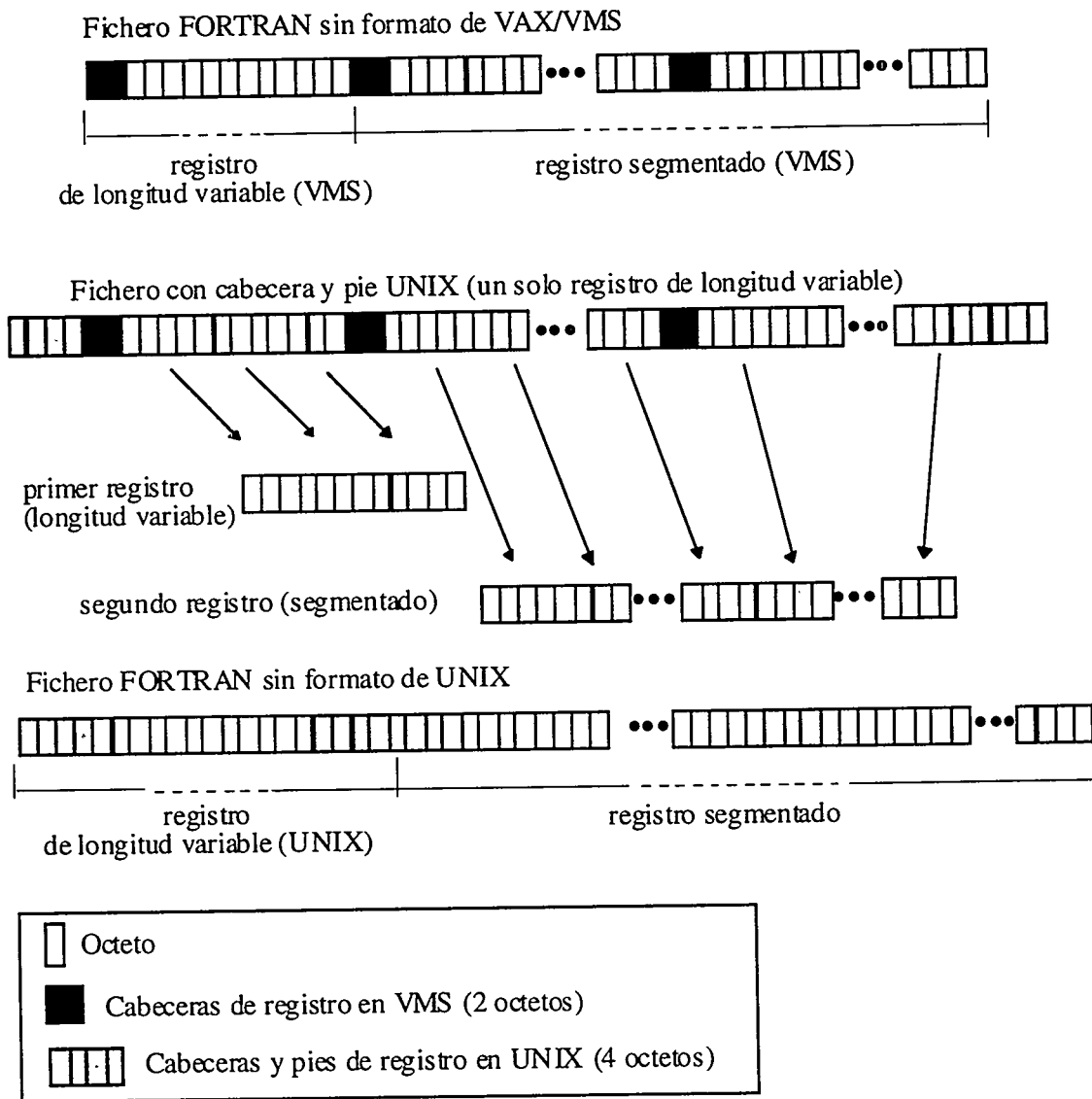


Figura 1. Visión esquemática de la conversión realizada por `getvax` y `mgetvax`

Forma de uso de los procedimientos `getvax` y `mgetvax`

Para el uso de `getvax` y `mgetvax` se utiliza el FTP automático, con el fichero `.netrc`⁸ de UNIX. Se necesita dar el nombre o dirección IP del servidor VAX/VMS desde el que se desea transferir los datos, el directorio dentro del servidor VAX/VMS en el que se encuentran los ficheros, y el nombre final que se quiere dar al fichero en el servidor ALPHA/Digital UNIX.

La sinopsis es la siguiente:

```
getvax vax.serv.red "dir_vax" "nom_fich" "nuevo_nom_fich"
```

⁸ Ver información sobre FTP y `.netrc` en el manual de UNIX

mgetvax vax.serv.red "dir_vax" "nom_fich"

donde,

vax.serv.red es la dirección IP o nombre del servidor VAX/VMS desde el que se desea transferir los ficheros. En el fichero **.netrc** del directorio del usuario ha de estar especificada esa dirección IP o nombre de máquina, así como el nombre de usuario y clave de una cuenta en el servidor VAX/VMS, con permisos de lectura para el(los) fichero(s) que se desea transferir.

dir_vax es el nombre del directorio del servidor VAX/VMS en el que se encuentran los ficheros a transferir.

nom_fich es el nombre del fichero a transferir. Con **mgetvax** se pueden utilizar asteriscos (*) para referirse a todos los ficheros cuyo nombre corresponda al patrón *nom_fich* conteniendo cualquier carácter o caracteres en el lugar del asterisco⁹.

nuevo_nom_fich es el nombre que se quiere dar al fichero una vez transferido (sólo con **getvax**). Si se omite este campo, el fichero transferido tendrá el mismo nombre que el original.

Una vez transferidos los ficheros y convertidos con estas herramientas, los ficheros se escribirán en el directorio desde el que se inició la operación

Para ayudar en el uso de estos procedimientos, se han incluido páginas **man** con información acerca de estos procedimientos, así como páginas referentes a los tipos DATU, DATRU y CATA definidos en el Apéndice B.

⁹ Son recomendables las comillas dobles para evitar que el shell de UNIX interprete los caracteres especiales, como los asteriscos

Transferencia automática de ficheros CAMAC, DATU, DATRU y CATA desde código FORTRAN

Los procedimientos **getvax**, **mgetvax** y **getcama** permiten transferir ficheros de forma cómoda para el usuario. Su ejecución es desde la línea de comandos de UNIX, aunque se podrían llamar desde código FORTRAN por medio de una llamada al sistema.

Para facilitar su uso desde código FORTRAN, se ha desarrollado una rutina (**getvaxfile**) integrada también en la librería **libcamac**, susceptible de ser enlazada con código FORTRAN y que a su vez llama a los procedimientos **getvax** o **getcama** según convenga.

Por medio de una llamada a esta rutina desde un programa FORTRAN se consigue que el fichero deseado sea transferido desde el servidor VAX al servidor ALPHA. Una vez hecho esto, si el fichero es de tipo CAMAC se pueden leer sus datos llamando a la rutina **camac_signal**, o bien, si el fichero es de tipo DATU, DATRU o CATA sus datos se pueden leer según los ejemplos del Apéndice B incluyendo en la apertura del fichero la opción **CONVERT='VAXD'[2,3]**.

Una vez que los datos deseados se han cargado a memoria, el fichero puede ser borrado (desde el propio código FORTRAN) con una llamada a la rutina **rmvaxfile**.

La forma de usar las rutinas **getvaxfile** y **rmvaxfile** es la siguiente:

getvaxfile (*servidor*, *directorio*, *fichero*, *tipo*)

servidor Cadena de 30 caracteres que identifica del servidor VAX/VMS en el que reside el fichero que se desea transferir. En el caso de un fichero CAMAC esta será simplemente el dec

directorio Cadena de 60 caracteres. Directorio en el que se encuentra el fichero a transferir, dentro del servidor VAX/VMS. En el caso de ficheros CAMAC, el fichero se buscará en el directorio de datos de TJI/TJ-IU independientemente de lo que aquí se ponga.

fichero Cadena de 30 caracteres con el nombre del fichero a transferir.

tipo Cadena de cinco caracteres que identifica el tipo de fichero a transferir. Los casos posibles son: '.DATU', '.DATR', '.CATA' o 'CAMAC' (¡ ha de incluirse el caracter '.' !).

A la rutina **rmvaxfile** hay que pasarle un único parámetro que es el nombre del fichero a eliminar, una cadena de 30 caracteres.

Apéndice A. Notas referentes a los formatos de los datos

Hay que hacer notar que, tanto para los ficheros FORTRAN sin formato como para los ficheros CAMAC transferidos desde un servidor VAX/VMS, las herramientas desarrolladas permiten el acceso a los datos desde código FORTRAN haciendo uso de las extensiones que el DEC FORTRAN para Digital UNIX proporciona para leer datos escritos en VMS (Véanse manuales de FORTRAN).

Se podrían haber convertido los datos a “formato UNIX” de modo que el acceso no necesitara el uso de dichas extensiones, y estos ficheros pasaran a formar parte del sistema de ficheros del servidor DEC 8400 de forma definitiva.

La razón de que no se haya realizado esa conversión, es que se pretende evitar la duplicidad de información que ya está en los servidores VAX en el servidor ALPHA.

La idea que movió el desarrollo de estas herramientas es el acceso a los datos para su análisis, permitiendo que los ficheros puedan residir en el servidor ALPHA temporalmente, ya sea el tiempo necesario para cargar los datos requeridos en memoria para su procesado, o bien un tiempo más largo si se requieren varios procesados sucesivos, pero, **en ningún caso, se deben dejar esos datos en el servidor ALPHA de forma definitiva**, ya que están almacenados en los servidores VAX y el espacio de disco del DEC 8400 se necesitará a buen seguro para dar almacenamiento a los datos tomados en el TJ-II.

Apéndice B: Tipos DATU, DATRU y CATA

En este apéndice se describe la estructura de tres tipos de ficheros **FORTRAN sin formato (DATU, DATRU y CATA)** ampliamente utilizados en la Unidad de Fusión para el almacenamiento de datos de maquinas de otros laboratorios con los que esta Unidad tiene colaboración.

Este tipo de ficheros es útil para el almacenamiento en una forma compacta de trazas temporales de señales con un periodo de muestreo fijo.

El tipo DATU

Los ficheros de este tipo contienen dos registros. El primer registro es de “longitud variable”, y el segundo es “segmentado”.

El primer registro contiene tres números:

- El primero es un real codificado en 4 octetos y representa el valor del tiempo de la primera muestra de la señal que está almacenada en el segundo registro.
- El segundo es un número real codificado en 4 octetos que representa la diferencia de tiempo entre dos muestras consecutivas de la señal: periodo de muestreo.
- El tercero es un número entero codificado también en 4 octetos y representa el número de muestras de señal que contiene el fichero a continuación

El segundo registro del fichero contiene las muestras de la señal, almacenadas una tras otra como enteros de 4 octetos.

A continuación se muestran dos ejemplos de código FORTRAN para escribir y leer respectivamente este tipo de ficheros.

Ejemplo 1w)

```
PARAMETER      (NDIM=1000)
INTEGER*4  NPUN
REAL*4      DELTA, TINI
INTEGER*4  TRAZA(NDIM)
(...)
OPEN( UNIT=1, FILE= ' FILENAME.DATU',
+      STATUS='NEW', FORM='UNFORMATTED')
WRITE(1) TINI, DELTA, NPUN
WRITE(1) (TRAZA(I), I=1, NPUN)
CLOSE(1)
```

(...)

Este ejemplo escribe 1000 muestras de la señal "TRAZA" en un fichero sin formato de tipo DATU llamado *filename.datu*.

Ejemplo 1r)

```
PARAMETER      (NDIM=1000)
INTEGER*4  NPUN
REAL*4      DELTA, TINI
INTEGER*4  TRAZA(NDIM)
(...)
OPEN( UNIT=1, FILE= ' FILENAME.DATU',
+      STATUS='OLD', FORM='UNFORMATTED')
READ(1) TINI, DELTA, NPUN
READ(1) (TRAZA(I), I=1, NPUN)
CLOSE(1)
(...)
```

Este ejemplo lee 1000 muestras de la señal "TRAZA" de un fichero sin formato de tipo DATU llamado *filename.datu*.

El tipo DATRU

Este tipo de fichero es similar al anterior, contiene también dos registros (uno de longitud variable y otro segmentado, justo en ese orden), con la diferencia de que el segundo registro contiene las muestras de la señal codificadas como reales de 4 octetos.

La escritura y lectura de datos según este formato es como en los siguientes ejemplos:

Ejemplo 2w)

```
PARAMETER      (NDIM=1000)
INTEGER*4  NPUN
REAL*4      DELTA, TINI
REAL*4      TRAZA(NDIM)
(...)
OPEN( UNIT=1, FILE= ' FILENAME.DATRU',
+      STATUS='NEW', FORM='UNFORMATTED')
WRITE(1) TINI, DELTA, NPUN
WRITE(1) (TRAZA(I), I=1, NPUN)
CLOSE(1)
(...)
```

Este ejemplo escribe 1000 muestras de la señal "TRAZA" en un fichero sin formato de tipo DATRU llamado *filename.datru*.

Ejemplo 2r)

```
PARAMETER          (NDIM=1000)
INTEGER*4  NPUN
REAL*4          DELTA, TINI
REAL*4  TRAZA(NDIM)
(...)
OPEN( UNIT=1, FILE= ' FILENAME.DATRU',
+     STATUS='OLD', FORM='UNFORMATTED')
READ(1) TINI, DELTA, NPUN
READ(1) (TRAZA(I), I=1, NPUN)
CLOSE(1)
(...)
```

Este ejemplo lee 1000 muestras de la señal "TRAZA" de un fichero sin formato de tipo DATRU llamado *filename.datru*.

El tipo CATA

El tipo CATA es muy similar a los dos anteriores, la diferencia es el orden de los números almacenados en el primer registro y la codificación de las muestras de la señal (enteros de dos octetos). La necesidad de su definición se debe a la gran cantidad de datos de TJ1-U almacenados en esta forma.

El primer registro (de longitud variable) contiene:

- número de muestras (entero de 4 octetos)
- tiempo de la primera muestra (real de 4 octetos)
- periodo de muestreo (real de 4 octetos)

y en ese orden.

El segundo registro (segmentado) contiene las muestras de la señal codificadas como enteros de 2 octetos.

Los siguientes ejemplos escriben y leen respectivamente datos en ficheros de acuerdo a este tipo:

Ejemplo 3w)

```
PARAMETER          (NDIM=1000)
```

```

INTEGER*4 NPUN
REAL*4      DELTA, TINI
INTEGER*2 TRAZA(NDIM)
(...)
OPEN( UNIT=1, FILE= ' FILENAME.DATU',
+     STATUS='NEW', FORM='UNFORMATTED')
WRITE(1) NPUN, TINI, DELTA
WRITE(1) (TRAZA(I), I=1, NPUN)
CLOSE(1)
(...)
```

Este ejemplo escribe 1000 muestras de la señal "TRAZA" en un fichero sin formato de tipo CATA llamado *filename.datu*.

Ejemplo 3r)

```

PARAMETER      (NDIM=1000)
INTEGER*2 NPUN
REAL*4         DELTA, TINI
INTEGER*4 TRAZA(NDIM)
(...)
OPEN( UNIT=1, FILE= ' FILENAME.DATU',
+     STATUS='OLD', FORM='UNFORMATTED')
READ(1) NPUN, TINI, DELTA
READ(1) (TRAZA(I), I=1, NPUN)
CLOSE(1)
(...)
```

Este ejemplo lee 1000 muestras de la señal "TRAZA" de un fichero sin formato de tipo CATA llamado *filename.datu*.

Referencias

- [1] CAMAC SOFTWARE FOR TJ-I AND TJ-IU, Milligen ,B. Ph. van
- [2] DEC FORTRAN, Lenguaje Reference Manual
- [3] DEC FORTRAN, User Manual for DEC OSF/1 ALPHA Systems.
- [4] VMS, VAX User's Manual
- [5] VAX FORTRAN, Lenguaje Reference Manual