

NOTAS TECNICAS SOBRE EL SISTEMASOBRE EL LENGUAJE APL

Los intercambios de información, exigen el empleo de lenguajes de la máxima precisión o exactitud, y si el discurso es demasiado largo, el uso de abreviaturas que posean las mismas cualidades,

Esta exigencia, tiene especial relevancia en los intercambios de información hombre-máquina. De ahí el nacimiento de los lenguajes de programación.

El problema de la implementación de un lenguaje de programación (no siempre de orden conceptual, sino más bien de orden tecnológico), ha hecho que pocos lenguajes de programación sirvan al mismo tiempo para la descripción de un sistema, y su tratamiento.

Un serio intento para resolver esta situación, fue el de K. IVERSON que desarrolló en 1962 un lenguaje con el nombre de " A Programming Language ", que posteriormente fue implementado bajo el nombre de A.P.L. (de hecho el APL es un subconjunto compilable del lenguaje original).

El APL es por tanto una notación formal, aplicable -

para expresar algoritmos matemáticos. Además, la notación es también no matemática, en el sentido de que problemas considerados no matemáticos, pueden ser expresados en APL.

El APL puede manejar conjuntos de caracteres, de la misma forma que lo hace con conjuntos de números, ésto permite que se pueda utilizar en edición de textos, análisis financiero, enseñanza asistida por ordenador, etc. En resumen, el APL es aplicable siempre que procedimientos no ambiguos, necesiten formalmente ser expresados.

Como cualquier buena notación, es util en diseño, - análisis y exposición, independientemente de la posibilidad de su ejecución por un ordenador. Sin embargo, ésto es importante cualquiera que sea la actividad que estemos desarrollando, y es una ventaja el que sea posible ejecutar algoritmos en la misma notación que empleamos para desarrollarlos.

Cuando se implementa sobre un ordenador, el APL es -más que una notación. Se convierte en un sistema de computación sobre el que el usuario crea y ejecuta funciones, y las almacena para usos posteriores.

La implementación puede estar hecha de muy diversas -formas, variando desde el proceso en " batch " hasta el tiempo compartido, basado este último en el uso interac

tivo desde terminales remotos conectados a una unidad central de procesamiento, de tal forma que un usuario puede estar geográficamente alejado de otros usuarios y del ordenador. La conexión se hace por medio de una línea privada, o mediante el uso de la red pública de telecomunicaciones, por ejemplo el teléfono.

El tiempo de ordenador está dividido en pequeños intervalos, cada uno asignado a un usuario, que, debido a la gran diferencia entre el tiempo humano y el tiempo máquina, tendrá la impresión de ser el único usuario del ordenador.

Dos versiones que actúan de esta forma son el APL/360 y el APLSV.

El APLSV añade a las características del APL/360, algunas funciones primitivas, y además provee de unos canales de comunicación entre programas que estén funcionando en dos terminales diferentes, mediante variables " compartidas " por ambos programas.

También un procesador auxiliar (TSIO) permite al usuario muchas de las facilidades, para tratamiento de datos, del sistema operativo, como control de la impresora-rápida, tratamiento de ficheros, etc.

CARACTERISTICAS DE LA NOTACION APL

El APL es una extensión de la notación algebraica en tres sentidos principales:

- La extensión de las funciones a arrays, es sistemática. Los arrays pueden tratarse como un todo.

- Provee un método formal para definir nuevas funciones.

- Provee un método formal para especificar la secuencia en la que una colección de sentencias, va a ser ejecutada.

Cada operación básica, o " función primitiva " va a ser denotada por un símbolo y sólo uno. Esta economía es vital para un sistema con un gran número de símbolos. Así, la potencia tiene asignado el símbolo* y por tanto no se escribirá A^B sino $A*B$. La división corresponde al símbolo \div y $A \div B$ es la división de A por B, y no es lo mismo A/B que tendrá otro significado. $A*B$ denota el producto y no equivale a $A.B$ ni a AB .

Las variables tienen un nombre de cualquier longitud y pueden referirse indistintamente a constantes vectores o arrays.

Una expresión que contenga símbolos de APL debe obedecer a un simple conjunto de reglas de sintaxis, como son:

lugar del símbolo: Si una función tiene dos argumentos (diádica), el símbolo que la denota se coloca entre ellos. Si la función tiene un único argumento (monádica), el símbolo se coloca delante. Por ejemplo el máximo de X e Y se escribe $X \uparrow Y$, el valor negativo de A será $-A$.

Reglas de asociación: En una expresión compuesta, - los paréntesis se utilizan, para determinar el orden de ejecución, en la forma usual. Dentro del paréntesis, o en ausencia de él, el orden de ejecución está sujeto a una única regla: toda función, toma como argumento a la derecha, el valor de toda la expresión que tiene a la derecha. No hay por tanto jerarquías entre las funciones. Por ejemplo la expresión $A+B \times C+D$ equivale a la expresión $A+(B \times (C+D))$

Notación de función: La aplicación de la " función - definida " F al argumento X puede escribirse FX en lugar de $F(X)$, aunque esta última forma es también válida, ya que están permitidos los paréntesis redundantes. En resumen, las características más sobresalientes del APL como lenguaje de programación son:

- Una sintaxis simple y consistente, aplicada a todas las funciones.
- Reglas simples para poder aplicar las funciones a argumentos que sean arrays.

- Un extenso conjunto de funciones primitivas que incluye funciones matemáticas, lógicas y otras para el tratamiento de arrays.

- La posibilidad de extender el lenguaje, definiendo nuevas funciones en términos de otras previamente definidas y de las primitivas.

La función del usuario de APL, consiste pues, en extender el lenguaje en orden a llegar a las funciones particulares que requiera su trabajo, y ejecutarlas para los argumentos que elija.

El usuario sólo necesita conocer el comportamiento de sus funciones. Nada más. Es posible, hacer uso extensivo del ordenador, sin un conocimiento relevante de cómo funciona.

Cuando el usuario da una expresión, el ordenador la evalúa, y cuando da una definición de función, el ordenador la almacena en memoria. Es tarea de la máquina el seguir la pista de nombres y sus significados, asignar espacio, identificar tipos de datos etc. También lo es el detectar instrucciones inválidas y permitir su corrección, sin abandonar el trabajo en curso.

EL APL EN NUESTRA INSTALACION

La versión implementada de APL instalada en el Centro de Cálculo, es el APLSV (APL Shared Variables). Está-
construído sobre la noción de " workspace " o zona de -
trabajo. Un usuario trabajando en un momento determina-
do, tiene asignado un workspace llamado activo, parte -
del cual es utilizado por el sistema para realizar la -
tarea del usuario, y el resto, donde almacenar las fun-
ciones definidas, variables y otra información generada
en el transcurso de la sesión,

El tamaño de un workspace en el APLSV del Centro de -
Cálculo es de 60 K-octetos,

El usuario puede poner un nombre a su wokrspace acti-
vo y almacenarlo en una biblioteca para utilizarlo pos-
teriormente,

Las bibliotecas pueden ser privadas o públicas. En -
una biblioteca privada, conocida por el número de iden-
tificación del usuario a quien pertenece, sólo él puede
almacenar o cambiar información.

En las bibliotecas públicas (con un número del 1 al-
999) , todos los usuarios pueden guardar información -
que consideren util al conjunto de usuarios de la insta-
lación, aunque sólo el que la almacenó, puede posterior-
mente modificarla.

Las operaciones para controlar los workspaces y las bibliotecas se realizan mediante comandos de un sencillo lenguaje de comandos que posee el APLSV.

Mediante comandos del lenguaje de comandos, el usuario puede realizar las operaciones de:

- iniciar y terminar una sesión de APL
- controlar el estado del workspace activo
- almacenar un workspace en una biblioteca, así como obtener una copia (de todo el workspace, o de parte de él) de una biblioteca o destruirlo.
- asignar un nombre a un workspace que va a ser almacenado, y obtener información de su contenido, como pueden ser las funciones en él definidas, las variables, y otras.
- enviar mensajes al operador o a otros terminales conectados al sistema.

Una protección contra el uso no autorizado de un número de cuenta o de los workspaces almacenados en una biblioteca privada, está prevista mediante el uso adecuado de palabras-clave.

El APLSV incluye la facilidad de compartir variables entre dos workspaces activos o entre un workspace activo y el procesador auxiliar TSIO que va a permitir crear, borrar o actualizar conjuntos de datos sobre cualquier dispositivo de almacenamiento del sistema operativo así como cambiar en un conjunto de datos, el nombre



bre, la longitud de sus registros o de sus bloques etc. o transferirlos de un dispositivo a otro.

Esto permite el obtener, en el workspace activo, datos sin necesidad de entrarlos desde el teclado del terminal, sino desde una unidad de cinta magnética, disco o lectora de tarjetas, y así mismo enviarlos, después de su tratamiento a unidades de cinta, disco perforadora de tarjetas, o a ser escritos por la impresora rápida. Operaciones todas ellas de gran interés, cuando se manejan grandes cantidades de datos que tardarían mucho en ser escritos desde o por el terminal, debido a su lentitud, o que no pudiesen ser almacenados en un workspace debido a su capacidad limitada de 60 K.

También se hace posible el enviar trabajos desde un terminal escritos en un lenguaje distinto del APL, para ser procesados en batch.

En la actualidad, hay cuatro terminales conectados al sistema, todos ellos situados en el edificio del Centro de Cálculo, desde los que es posible la utilización del APLSV por todos los usuarios del Centro, con el horario de 9'30 a 11'30 y de lunes a viernes.