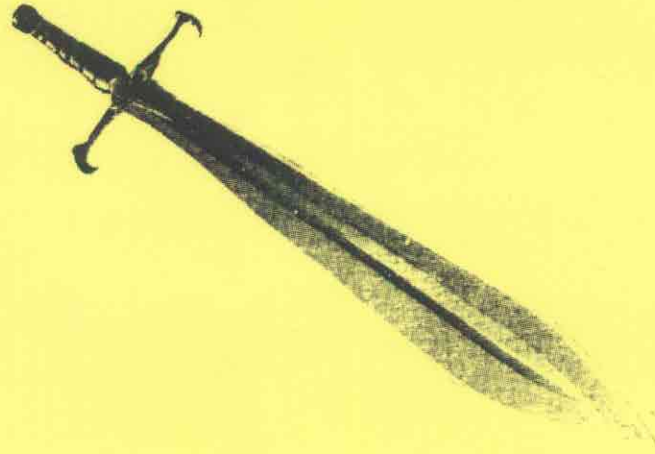


# The Professional Adventure Writing System

Guía Técnica



© 1986 Gilsoft International Ltd.

Published by Gilsoft International Ltd.,  
2 Park Crescent, Barry, South Glamorgan. CF6 8HD  
Telephone: Barry (0446) 732765

Distribuidor Oficial para España : A.D.S.A.

All rights reserved, unauthorised copying, hiring or lending strictly prohibited

DOALL locno+

;Genera el Nombre(Adjetivo)1 para cada objeto que esté en la localidad marcada por locno.

RESET Tipo 1 locno.

;Mueve al jugador y a los objetos presentes. Resetea los demás a la posición inicial (es el usado en juegos de partes múltiples con el comando LOAD, ver el ensayo correspondiente).

EXTERN 0-255

;Hace una llamada a un programa externo

## SISTEMA PROFESIONAL PARA LA CREACION DE AVENTURAS GRAFICAS

Versión en Castellano

(c) programa: 1986/89 Gilsoft International Ltd.

(c) versión castellana: Aventuras A.D., S.A.

### DERECHOS DE TRADUCCION, ADAPTACION Y REPRODUCCION RESERVADOS

Prohibida la reproducción total o parcial del sistema

Los juegos producidos con PAW, quedan libres de toda restricción, agradeciéndose se cite la fuente.

Pone un número aleatorio de 0 a 99  
Ajusta el contenido de la bandera de  
acuerdo al Verbo de la SL y de la  
entrada en la tabla de conexiones  
para las localidades. (controla el  
movimiento de un PSI)

Introducción	Pag. 5
El Intérprete	Pag. 7
1. Inicialización	Pag. 7
2. Descripción de localidad actual	Pag. 7
3. Tabla de Procesos 1	Pag. 8
4. Tabla de Procesos 2	Pag. 8
5. Coger la frase	Pag. 8
6. Tabla de Respuestas	Pag. 9
7. Tabla de Conexiones	Pag. 10
Como se busca en una tabla de Procesos	Pag. 13
Los Conductos	Pag. 15
Condiciones	Pag. 15
Acciones	Pag. 20
Descripción detallada de la base de datos	Pag. 48
Descripción detallada del Editor	Pag. 56
Descripción detallada del Editor Gráfico	Pag. 74

SUB		flagno1			
COPYFF		flagno1		flagno2	Pone un número aleatorio de 0 a 99
RANDOM		flagno			Ajusta el contenido de la bandera de acuerdo al Verbo de la SL y de la entrada en la tabla de conexiones para las localidades. (controla el movimiento de un PSI)
MOVE		flagno			
GOTO		locno.			
WEIGHT		flagno			El peso de los objetos llevados o puestos encima se ponen en la bandera
ABILITY		0-255		0-255	Pone los objetos llevados y la fuerza
MODE		0-4		0-3	
LINELINE		0-20			Pone línea de pantalla para los juegos que llevan dibujo y texto
GRAPHIC		0-3			
PROMPT		sysno			Ejecuta un Prompt en el input. Si se utiliza 0 es aleatorio
INPUT		0-7			
TIME		0-255		0-7	
PROTECT					Protege el texto en la pantalla desde la línea de impresión actual del scroll
PRINT		flagno			Muestra el contenido de la bandera en la pantalla
TURNS					
SCORE					
CLS					
NEWLINE					
MES		mesno			mensaje sin dejar línea en blanco
MESSAGE		mesnc			mensaje con línea en blanco
SYSMESS		sysno			mensaje del sistema sin línea en blanco
PICTURE		locno			
PAPEL		0-9			Muestra el dibujo (sin CLS)
TINTA		0-9			

## Condiciones:

AT	locno.	;Se asegura de que el jugador esté en una localidad especifica.
NOTAT	locno.	
ATGT	locno.	;una localidad mas alta que la especificada.
ATLT	locno.	;una localidad mas baja...
PRESENT	objno	
ABSENT	objno	
WORN	objno	
NOTWORN	objno	
CARRIED	objno	
NOTCARR	objno	
ISAT	objno	locno+
ISNOTAT	objno	locno+
ZERO	flagno	
NOTZERO	flagno	
EQ	flagno	0-255
NOTEQ	flagno	0-255
GT	flagno	0-255
LT	flagno	0-255
SAME	flagno	flagno
NOTSAME	flagno	flagno
ADJECT1	palabra	
ADVERB	palabra	
PREP	palabra	
NOUN2	palabra	
ADJECT2	palabra	
CHANCE	0-99	;posibilidad aleatoria de que suceda
TIMEOUT		;en el último input del jugador hay un Tiempo Muerto

QUIT

## INTRODUCCION

PAW puede ser dividido en tres partes principales:

### 1/ La Base de Datos

La base de datos es una colección de tablas organizadas de manera lógica, que contienen toda la información para definir un juego de aventuras.

Esta base de datos está cargada inicialmente con un mínimo de entradas que probablemente se necesitarán en cualquier juego.

Normalmente hay un entrada en cada tabla para simplificar la forma en que opera el sistema.

La base de datos puede utilizar páginas de RAM adicionales en un Spectrum de 128K.

### 2/ El Editor

Consiste en un sistema de menús en dos niveles (esto quiere decir que la mayoría de las opciones del menú principal presentarán un submenú del cual se pueden seleccionar otras opciones).

Todas las órdenes pueden entrar, o bien desde una línea de mando como parte de un submenú, o como respuesta a una pregunta del ordenador.

Esta combinación de menú y línea de mando la consideramos la más apropiada y eficiente tanto para el principiante como para el veterano.

### 3/ El Intérprete

Es el corazón real de PAW. En el siguiente capítulo trataremos de él con más detalle, pero debes saber que lo que hace de una manera general: recibe órdenes del jugador y usa la información contenida en la base de datos para decodificarla y responder a ellas.



- Bandera 50 Lleva el Número del Objeto para un bucle de DOALL.  
(Es el valor que sigue a DOALL).
- Bandera 51 Lleva el último objeto a que se refiere un GET/DROP/WEAR/WHAT, etc.  
Esto es, un número del Objeto a que se refiere actualmente, para ser impreso en el lugar de un      en el texto.
- Bandera 52 La fuerza del jugador (el máximo peso de los objetos llevados mas los puestos encima)  
Inicialmente es 10. (ABILITY)
- Bandera 53 Es la bandera para imprimir los objetos en INVENTARIO.  
Bit 7 (128) = Seteado para que cualquier objeto se imprima en un LISTOBJ o un LISTAT.  
Bit 6 (64) = Seteado para que los objetos se impriman en una línea continua. Por ejemplo, LET 53 64 hará que se listen en una sola línea formando una frase.
- Bandera 54 Lleva el número de la loc. del objeto actualmente referido.
- Bandera 55 Lleva el Peso del objeto actualmente referido.
- Bandera 56 Estará a 128 si el objeto actualmente referido es un contenedor.
- Bandera 57 Estará a 128 si el objeto actualmente referido se puede llevar puesto encima.
- Bandera 58 y 59 NO DEBEN USARSE PUES SE USARAN EN FUTURAS EXPANSIONES.
- Bandera 60- 255 ESTAN LIBRES PARA USAR.

## EL INTERPRETE

La descripción siguiente del Interpretador debe ser leída al mismo tiempo que se estudia la tabla de flujo de las páginas.

### 1-LA INICIALIZACION DEL SISTEMA

Los colores de fondo y el set de caracteres son seleccionados. La pantalla no se limpia puesto que siempre ocurrirá al describir la localidad 0.

Las banderas son todas puestas a 0 excepto:

La bandera 37, contiene el número de objetos llevados, que se pondrá en 4.  
La bandera 52, lleva el máximo peso permitido, que se pone a 10.  
Las banderas 46 y 47 que llevan el pronombre actual, que se pondrán en 255 (no hay pronombre).  
La bandera 1, que lleva el número de objetos llevados pero no usados.

Hay que tener en cuenta que con la limpieza de banderas, el juego siempre comenzará en la localidad 0. Esto es porque la bandera 38 que lleva a la localización actual del jugador está ahora en cero.

### 2-DESCRIPCION DE LA LOCALIDAD ACTUAL

Si la bandera 2 no está a cero será decrementada en 1.

Si está oscuro (que la bandera 0 no es cero) y la bandera 3 no es cero entonces la bandera 3 es decrementada. Si está oscuro y la bandera 4 no está a cero y el objeto 0 (la fuente de luz) está ausente, la bandera 4 es decrementada.

Se hace una limpieza de pantalla (clear) si el contenido del modo de pantalla (el que se contiene en la bandera 40) no está a 1.

Si está oscuro y el objeto 0 está ausente, entonces el Mensaje del Sistema 0 (el que se refiere al mensaje "está muy oscuro para ver") se imprime.

- Bandera 30 Lleva la puntuación.
- Bandera 31 y 32 (LSB/MSB) Turnos o número de frases del input del jugador.
- Bandera 33 El Verbo de la SL actual.
- Bandera 34 El primer Nombre de la SL actual.
- Bandera 35 El Adjetivo del primer Nombre.
- Bandera 36 El Adverbio de la SL actual.
- Bandera 37 El número máximo de objetos llevables. (ABILITY)  
Por defecto = 4
- Bandera 38 La localidad actual.
- Bandera 39 La línea superior del texto. (LINEA)
- Bandera 40 Controla el display de pantalla. (MODE)  
0 = Si hay dibuj. en base de datos (no subrutina)  
se hace un CLS a los "Colores por Defecto" y se se dibuja. Luego, al apretar tecla = se restituyen los colores originales y se hace DESC.  
1 = Sólo Texto. Hay Scroll completo de pantalla. CLS no está activado al DESC.  
2 = Si hay dibujo en base de datos (no subrutina) se hace un CLS a los "Colores por Defecto" y se se dibuja. Los colores originales se restituyen, el área bajo la línea dada por Ban 41 hace un CLS y se imprime texto desde esa misma línea.  
3 = Como en dos. Pero el dibujo no hace Scroll.  
4 = Para Texto sólo. Pero protegiendo el DESC como si fuera dibujo.  
Bit 7 (128) fuerza el que no haya cambio de Border  
Bit 6 (64) produce el mensaje "Hay más.." cuando se llena la pantalla.
- Si el buffer de input esta vacío, se busca un nuevo input del jugador por medio de la impresión de la frase adecuada y una posterior llamada a la rutina de input. El mensaje será el que contenga la bandera 42. Si tiene un valor igual a 0 los mensajes 2,3,4 y 5 serán seleccionados con una frecuencia de 30:30:30:10 respectivamente.
- Si se selecciona la opción tiempo muerto (timeout), haciendo que la bandera 48 tenga un valor más grande que 0, entonces se ejecutará durante la rutina de input. En este caso el mensaje del sistema MS35 "El tiempo pasa..." aparece y se vuelve otra vez a la búsqueda en la tabla de Procesos 2.
- Una frase es sacada y convertida en una sentencia lógica por medio de la conversión de cualquier palabra en ella presente, que esté en el vocabulario, a su número de palabra y poniéndola luego en la bandera requerida.
- Si no se encuentra una frase válida, entonces el mensaje del sistema MS6 "No entiendo" aparece y vuelve a buscar al Proceso 2.

#### 6- BUSQUEDA EN LA TABLA DE RESPUESTAS

El turno es llevado por las banderas 31 y 32, y es el número de frases válidas que han sido sacadas por el parser. Este turno entonces se aumentará en 1.

Se usan dos banderas para permitir un total de 65.536 turnos, 256 por 256, (si el jugador no ha resuelto el juego para entonces, es porque está tarado).

En la tabla de Respuestas se busca una entrada que haga juego con el Verbo y el primer Nombre de la Sentencia Lógica actual.

Si es positivo se lleva a cabo la acción (excepto si la acción es NOTDONE) y luego se vuelve a la búsqueda en la tabla de Procesos 2.



MS7 - es producido si no se lleva a cabo ninguna acción (o una acción NOTDONE fue ejecutada) en la tabla de Respuesta cuando el Verbo sea menor > 14.

MS8 - se produce si no se lleva a cabo una acción (o la acción NOTDONE fue ejecutada) en Respuesta si el Verbo es mayor < 13.

MS9 al MS11 - se imprimen por medio de la acción INVEN.

MS12 - impreso por QUIT (abandonar).

MS13 y MS14 - impresos por la acción END.

MS15 - el mensaje que responde a la acción OK.

MS16 - el mensaje de la acción ANYKEY.

MS17 al MS20 - son mensajes que corresponden a la acción TURNS.

MS21 y MS22 - mensajes que corresponden a la acción SCORE.

MS23 al MS29 - son los primeros de los muchos mensajes producidos por acciones que manipulan los objetos.

MS30 - la respuesta positiva que se espera por la acción END y QUIT.

MS31 - la respuesta negativa que se espera por la acción END y QUIT.

MS32 - se produce cuando aparece una pantalla llena de texto.

MS33 - es donde se marca el input.

MS34 - el cursor.

MS35 - aparece mientras se toma un tiempo muerto.

MS36 al MS45 - son más mensajes producidos por las acciones que manipulan los objetos.

MS46 - es el lazo de unión entre objetos cuando se están listando continuamente.

MS47 - es el enlace final entre los dos últimos objetos cuando se está haciendo un listado.

MS48 - es la terminación de un listado de objetos. Se usa tanto para LISTOBJ como para LISTAT, así que tén cuidado con él, (generalmente suele ser un punto).

MS49 y MS50 - aún más mensajes para manejo de objetos.

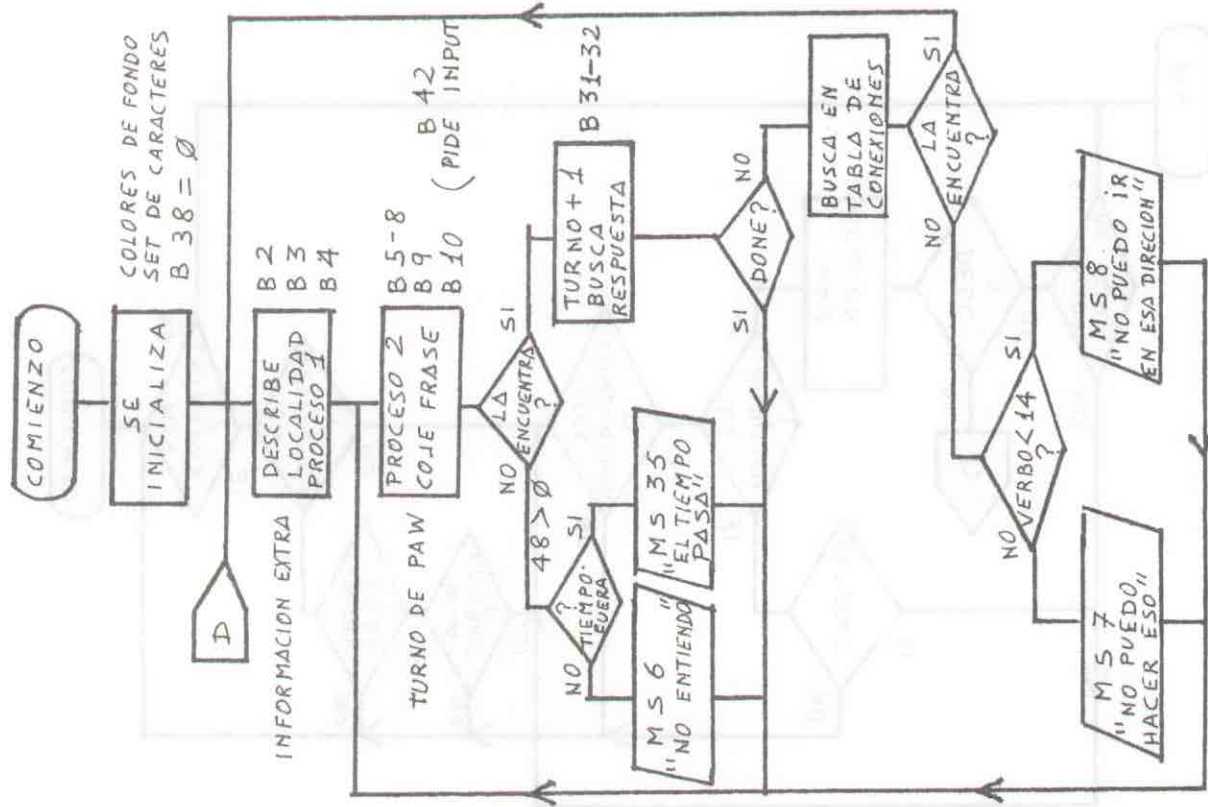
MS51 - es la terminación para una sentencia compuesta en PUTIN/TAKEOUT (y AUTOP/AUTOT).

MS52 - es un mensaje final de objetos.

MS53 - es el mensaje que aparece cuando durante una acción LISTAT no se encuentran objetos.

Desde MS54 hacia arriba, están libres para tu propio uso. Pero PAW en otras máquinas futuras puede usar más mensajes, esto hay que tenerlo en cuenta si se intenta en el futuro transferir la aventura a otra versión. Por ejemplo, PAW en CP/M usa los mensajes 54 a 60 (pero PAW en CP/M de momento no lleva gráficos).

Tabla de Flujo 1



## 1. ORDENES DEL EDITOR GRAFICO

Ocasionalmente notaremos que al mover el cursor a velocidad rápida puede desaparecer; esto puede ser debido a que la frecuencia de barrido del monitor coincide con la velocidad de impresión de la línea en pantalla.

Se mueve el final de la banda elástica con:

Q W E un pixel por movida. También se puede usar  
A D el Interface 2/Plus 2, joystick port 2,  
Z X C o el Interface Kempston (SIMBOL SHIFT + J)

Se puede acelerar el movimiento de 8 en 8 pixels con CAPS SHIFT.

Los comandos de edición:

Cursor Derecha (CAPS SHIFT & 8) - Comienzo de la línea de dibujo  
Cursor Abajo (CAPS SHIFT & 6) - Nueva orden  
Cursor Arriba (CAPS SHIFT & 7) - Comando previo

DELETE (CAPS SHIFT & 0) - Borra el comando previo  
GRAFICOS (CAPS SHIFT & 9) - Borra el siguiente comando

Los siguientes comandos requieren SIMBOL SHIFT a menos que se especifique lo contrario:

I Causa un Inverse  
O El Over  
Y El Grid, o sea, la Reticula  
J El joystick Kempston

Todos pueden estar encendidos o apagados si se teclean una o dos veces.

## BUSQUEDA EN UNA TABLA DE PROCESOS

La tabla de Respuesta es una tabla de Procesos (y cualquier sub-proceso que se llame desde la tabla de Respuestas recibirá el mismo tratamiento), por lo cual siempre será buscada en esta sección de PAW. La diferencia es, por supuesto, que PAW intenta buscar una similitud entre el Verbo y el Nombrel de la SL y la tabla de Respuestas.

Pero esencialmente, lo que PAW hace es mirar una por una cada entrada de la tabla, hasta que se acabe (no PAW sino la tabla). Si encuentra una entrada en la tabla de Respuestas se asegurará de que el Verbo y el Nombre sean los mismos a los de la SL.

Por ello el uso de palabras como "\*" y "-", bien sea como Verbo o como Nombre, hará que sea positiva esa palabra para cualquier parte de la SL. Por lo tanto, una entrada en la tabla de Respuestas como "\*", "-", o cualquier combinación de las mismas, disparará cualquier entrada, no importa lo que se haya tecleado.

Este hecho debe ser también usado en cualquier tabla que vaya a ser llamada, bien sea por una tabla de Respuestas o una tabla de Procesos. De todos modos, en una tabla de Procesos, el Verbo y el Nombre serán similares siempre a lo que teclees en la sentencia lógica.

PAW entonces va a examinar cada uno de los conductos. Si es una condición válida PAW buscará el siguiente conducto. Si no, se saldrá de la actual lista de conductos y buscará otra entrada en la tabla (una excepción de esta regla es el conducto QUIT, el cual, si falla, se saldrá completamente de la tabla). Esto no lo mostramos en la tabla de flujo 2 para una mayor claridad.

Si se trata de una acción, entonces se llevará a cabo.

Las acciones pueden ser divididas en cinco grupos principales:

1/ DESC; que causará una salida completa de todas las tablas (aunque esté en un sub-proceso de nivel 10) y saltará a describir la localidad actual.



Para poder hablar con el Archivero necesitamos dos entradas en la tabla de Respuesta, de la siguiente manera:

DECIR ARCHIVERO PREP A ;esto último (PREP A) puede ser omitido si solamente queremos que acepte la forma de hablar tipo indio "Decir Archivero", si ponemos PREP A también se puede decir "Decir al Archivero" (lógicamente, debes tener A y AL como sinónimos).

SAME 20 38 ;para asegurarnos de que el Archivero esté aquí

PROCESO 3 ;es el que se encarga de hablar

DONE ;esto se pone para impedir que la acción caiga a una nueva SL

DECIR - PREP A ;esto es opcional  
MESSAGE 2 ;que dice que no hay nadie con ese nombre  
DONE

Por su parte en la tabla de Procesos 3, hay que poner las siguientes entradas:

\* \* PARSE ;es la que convierte cualquier línea de "input" a una SL

MENSAJE 1 ;es donde caerá la acción si PARSE no encuentra una frase válida

DONE ;esto es porque la SL no es la adecuada y no se deben de ejecutar más entradas

HOLA - MENSAJE 3 ;suponiendo que HOLA sea verbo en el Vocabulario

DONE ;para que al decir "HOLA" al Archivero funcione

## LOS CONTACTOS

Ahora seguirá una descripción detallada de cada contacto. Están divididos en grupos, de acuerdo con el tema del que se encargan dentro del PAW; como banderas, objetos, etc. También damos algunas claves sobre su posible uso.

En las descripciones se usan varias abreviaturas que es importante conocer:

locno. es el número de una localidad válida.

locnot es un número válido de localidad:

252 (no creado)

253 (llevado puesto)

254 (llevado)

255, el cual se va a convertir en la localidad actual del jugador.

mesno. es un mensaje válido.

sismo. es un Mensaje del Sistema válido.

flagno. es cualquier bandera (de 0 a 255).

procno. es un número válido de sub-proceso.

palabra es una palabra del tipo requerido que está presente en el vocabulario, o " \_ " que ahora significa no palabra (que no significa cualquier palabra como hasta ahora).

valor es un valor de 0 a 255.

## CONDICIONES

Hay cuatro condiciones que tratan de probar la localidad del jugador:

AT locno.

Satisfactorio si la localidad actual es la misma que pone en locno.

La bandera debe de mostrar en todo momento donde está el personaje, los mensajes nos dan información sobre sus acciones y las tablas de Procesos son las que mueven todo el cotarro.

Así que imaginemos un personaje llamado "Archivero", que puede andar por todo el juego independientemente.

Hay que describirlo en el Vocabulario como ARCHIVERO (esperemos que por ahora ya sepas que el parser sólo aceptará las primeras cinco letras, pero ahora usaremos la palabra completa para facilitar la comprensión), y le damos un valor de 25 y la categoría de Nombre.

La bandera número 20 se usa para mantener su localidad.

Escogeremos la tabla de Proceso 3 para las entradas que tengan que ver con lo que se le habla y las Repuestas.

La tabla de Proceso 4 nos servirá para controlar sus movimientos y sus acciones.

Las siguientes entradas le permitirán a él moverse por todo el juego cuando así se lo pida el jugador.

Y haremos que después de algún tiempo el Archivero se aburra y desaparezca en una nube de humo.

Mensaje 1

El Archivero parece no entenderte.

Mensaje 2

No hay nadie con ese nombre por aquí.

Mensaje 3

El Archivero te dice "Hola" con una tenebrosa voz.

Mensaje 4

El Archivero va a su marcha, puesto que no encuentra ninguna cosa mejor que hacer.

**NOTCARR objno.**

Satisfactorio si el objeto que trata objno no se lleva.

**ISAT objno. locno+**

Satisfactorio si el objeto de que trata objno está en el locno de la localidad.

**ISNOTAT objno. locno+**

Satisfactorio si el objeto de que trata objno no está en el locno de la localidad.

Hay ocho condiciones que tratan de los valores y de la comparación de banderas:

**ZERO flagno.**

Satisfactorio si el número de la bandera indicado por flagno es igual a cero.

**NOTZERO flagno.**

Satisfactorio si el número de la bandera indicado por flagno no es igual a cero.

**EQ flagno. valor**

Satisfactorio si la bandera indicada por flagno tiene un valor igual al indicado.

**NOTEQ flagno. valor**

Satisfactorio si el número de la bandera indicada por flagno no es igual al valor.

**GT flagno. valor**

Satisfactorio si el número de la bandera indicado por flagno es mayor que el valor.



## 8. EL COMANDO EXTERN Y LA CARGA DE PANTALLAS

El comando EXTERN puede ser usado para llamar tus propios programas en código máquina o a una sección de un programa en BASIC. Esta opción puede ser utilizada solamente en el juego final, porque las 4K de memoria ocupadas previamente por el menú principal se harán usables. (Las otras 2K que quedan es el lugar de trabajo de PAW, así que ni tocarlo).

La dirección importante es el valor que se imprime cuando PAW se carga (llamado EXTVEC). Debes usar siempre el valor que se da en la copia de PAW desde la cual grabaste el juego final, o tendrás el caos asegurado.

### PARA LOS ENTENDIDOS EN CODIGO MAQUINA:

Para que el comando EXTERN pueda ejecutar rutinas de código máquina, debes de hacer un POKE de tres bytes en el EXTVEC con un JP hacia tu dirección de comienzo. El registro IX debe de ser preservado, puesto que está apuntando hacia la bandera 0. Todos los demás pueden ser destruidos, pero SP debe de ser preservado.

El registro A, de entrada contiene el parámetro hacia el comando EXTERN. Si grabas los dos bytes en EXTVEC + 1, ellos apuntarán hacia la rutina que llama el programa de BASIC. El número de línea debe de ponerse en A.

### PARA LOS USUARIOS DE BASIC:

Un POKE a EXTVEC, 195 (después de haber cargado el Intérprete) permitirá que cualquier acción EXTERN ejecute la línea de BASIC (100 + parámetros), por ejemplo, EXTERN 10 irá a la línea 110 en BASIC. Para volver al siguiente condacc debe de ponerse un comando de tipo STOP en BASIC.

No se puede usar el CLS en BASIC (o en código máquina), pero la dirección EXTVEC + 3 contiene una rutina que ya lo hace, por ejemplo: RANDOMIZE USR extvec+3 hará un clear de la pantalla.

Además, EXTVEC+12 contiene el número de línea donde se imprimen los nombres de las bases de datos y EXTVEC+9 contiene el comando que carga la base de datos y empieza el juego.

## ADJECT2 palabra

Satisfactorio si el adjetivo del segundo Nombre en la SL actual es igual a palabra.

La siguiente condición es utilizada para efectos al azar. Se puede usar para que las cosas ocurran como si fuera por casualidad.

Por ejemplo, la caída de un árbol sobre el jugador durante una tormenta, o un puente que se derrumba, etc. Recomendamos no abusar de esta facilidad y dar siempre al jugador un aviso que le ayude a prevenir el problema y a solucionarlo.

## CHANCE porcentaje

Satisfactorio si el porcentaje es menor o es igual a un número aleatorio entre 1 y 100 (ambos inclusive). Ello significa que una condición con CHANCE 50 permitirá a PAW seguir buscando en el próximo contacto solamente si el número aleatorio generado era entre 1 y 50, o sea, un porcentaje del 50% de que salga satisfactorio.

Una condición para probar el "tiempo muerto" o "fuera de tiempo":

## TIMEOUT

Será válida si la última petición de INPUT para el jugador fué permitida en fuera de tiempo, es decir, sin relación con lo que teclée.

Por ejemplo, una entrada en la tabla de Procesos 2 que diga TIMEOUT MESSAGE 0 (donde el mensaje 0 diga "venga, despierta") podría ser creada, quizá con un CHANCE o una cuenta de "fuera de tiempo" para hacerla menos monótona.



Hay que tener en cuenta que se debe insertar una acción de tipo CLEAR para las banderas entre la acción ANVKEY y RESET, puesto que las banderas no se ven afectadas por la orden RESET.

## 7. DIA Y NOCHE (LUZ Y OSCURIDAD)

La oscuridad se está haciendo muy común en las aventuras actuales, y de hecho, usada correctamente puede añadirle una sensación de realismo bastante fuerte al juego.

Dentro de PAW la oscuridad se crea SETEANDO LA BANDERA 0 A UN VALOR DIFERENTE DE 0. ES DECIR:

**BANDERA 0 IGUAL a 0 = LUZ**

**BANDERA 0 DIFERENTE a 0 = OSCURIDAD**

Estos cambios deben hacerse siempre que un jugador se mueva hacia o desde una zona de oscuridad, y este movimiento debe hacerse mediante una acción GOTO en la tabla de Respuestas, para permitir el poder SETEAR o CLEAR de la bandera.

Si el jugador tiene una fuente de luz, entonces esta fuente de luz SERA SIEMPRE EL OBJETO 0 (Fuente de luz = Objeto 0).

Una fuente de luz no tiene porque ser siempre una antorcha o una llama, si tienes un poco de imaginación, pueden ser unas gafas infrarrojas o un láser de luz amplia.

Vamos a tomar como ejemplo la creación de un ciclo de día y de noche, usando 24 espacios de tiempo (que asumimos que son equivalentes a 1 hora cada uno).

Las entradas requeridas en la tabla de Procesos 2, son:

CICLO	EQ	5	0	:Final del ciclo
LET	5	24	:Nuevo comienzo del contador	

## DROP objno.

1-Si el objeto a que se refiere objno se lleva puesto encima, entonces el Mensaje del Sistema 24 "No puedo. Llevo puesto \_" se imprime y las acciones NEWTEXT y DONE se ejecutan.

2-Si el objeto a que se refiere objno está en la localidad actual (pero ni se lleva puesto encima, ni es llevado), el Mensaje del Sistema 49 "No tengo \_" se imprime y las acciones NEWTEXT y DONE se ejecutan.

3-Si el objeto a que se refiere objno no está en la localidad actual, entonces el Mensaje del Sistema 28 "No tengo eso" se imprime y las acciones NEWTEXT y DONE se ejecutan.

4-Si no se dan los casos anteriores, la posición del objeto al que se refiere objno se cambia a la localidad actual. La bandera 1 se decrementa y el Mensaje del Sistema 39, "He dejado caer \_" se imprime.

## WEAR objno.

1-Si el objeto al que se refiere objno está en la localidad actual (pero no se lleva, ni está puesto encima) el Mensaje del Sistema 49 "No tengo \_" se imprime y las acciones NEWTEXT y DONE se ejecutan.

2-Si el objeto al que se refiere objno es llevado puesto encima, el Mensaje del Sistema 29 "Ya llevo puesto \_" se imprime y las acciones NEWTEXT y DONE se ejecutan.

3-Si el objeto al que se refiere objno no se lleva, el mensaje del sistema número 28 "No tengo uno de esos" se imprime y las acciones NEWTEXT y DONE se ejecutan.

4-Si el objeto al que se refiere objno no se ha marcado por medio de una bandera como "usable" (con la opción WR en el menú Pesos de Objetos), entonces el Mensaje del Sistema número 40 "No puedo ponerme \_" se imprime y las acciones NEWTEXT y DONE se ejecutan.

## 6. AVENTURAS MULTIPLES

Si deseas crear aventuras más largas, con un área mucho más grande de juego, sin sacrificar la calidad de la descripción, puedes dividir el juego en secciones mucho más pequeñas.

Esto es mucho más fácil si se ha planeado el juego con áreas de unión entre zonas. Es lo que se llama un cuello de botella.

Por ejemplo, un juego en el cual la primera parte termine en que debes de subirte a un barco y echarte a navegar. Así, el juego es mucho más fácil de dividir.

Para que la puntuación, los turnos que has hecho, y otra información, se puedan llevar de una parte del juego a la siguiente, debes usar acciones que te permitan CARGAR/GRABAR (load/save) las posiciones del juego.

Para poder CARGAR una posición de juego en un juego diferente de aquel en el cual fue GRABADO, necesitas tener el mismo número de localidades y de objetos en cada parte.

Además, todos los objetos que pueda llevarse el jugador de una parte a otra deben tener la misma descripción en todas las partes.

Supongamos que tenemos un juego de 120 localidades que vamos a partir por la mitad. Ello requiere 60 localidades en cada parte.

De hecho, la localidad 60 existirá en ambos juegos como localidad de transición (es decir, donde el jugador empieza la segunda parte y termina la primera) y también necesitamos una bandera libre, (digamos la 26), que será usada para indicar de qué parte del juego viene la posición que se ha GRABADO.

Así pues, cuando el jugador complete la primera parte, se mueve a la localidad 60 y la bandera 26 se setea a 1 para marcarlo.

### SWAP objno1 objno2

Las posiciones de dos objetos son cambiadas. La bandera 1 no se toca para nada. El objeto al cual se refiere actualmente, es decir, el último referido y que PAW conserva en memoria en la bandera 51, se sitúa en el objno número 2 (el que está marcado en la bandera 51).

### PLACE objno. locno+

La posición a la que se refiere el objno se cambia a la localidad a la que se refiere locno. La bandera 1 se decrementa si era un objeto llevado. Y se incrementa si el objeto se pone en la localidad 254 (llevada).

### PUTO locno+

La posición del objeto al que se refiere actualmente (el objeto a cuyo número esté en la bandera 51) se cambia para ser el número de localidad de locno. La bandera 54 permanece como su vieja localidad. La bandera 1 se decrementa si el objeto era llevado, y se incrementa si el objeto se pone en la localidad 254 (llevado).

### PUTIN objno. locno.

1-Si el objeto con número de locno. no existe o no está marcado como un contenedor (opción C en el menú de Objetos Peso) entonces se genera un error de "Argumento Ilegal".

2-Si el objeto al que se refiere objno. es llevado puesto encima, entonces el Mensaje del Sistema 24 "No puedo, llevo puesto \_" se imprime y las acciones NEWTEXT y DONE se ejecutan.

3-Si el objeto al que se refiere objno. está en la localidad actual (pero ni se lleva puesto encima, ni se lleva) el Mensaje del Sistema número 49 "No tengo \_" se imprime, y las acciones NEWTEXT y DONE se ejecutan.

4-Si el objeto al que se refiere objno. no está en la localidad actual, pero tampoco está llevado, entonces un Mensaje del Sistema número 28 "No tengo uno de esos" se imprime, y las acciones NEWTEXT y DONE se ejecutan.



Hay que hacer notar que la orden MES deja que el próximo mensaje se imprima justo después de él; es decir, no lleva implícito un NEWLINE como la orden MESSAGE.

## 5. OBJETOS

La raya baja en el texto será convertida durante el juego a la descripción del último objeto al que se refiera una acción COGER, DEJAR, DESTRUIR, etc.

Esto se hace principalmente por el hecho de que los Verbos COGER, DEJAR, etc., hacen una respuesta automática de su fallo o de su éxito. Por ejemplo, cuando tecleas "COGER CUCHILLO", el parser responde "YA HAS COGIDO EL CUCHILLO". Pero también puede ser usado para examinar objetos y multitud de otras funciones automáticas.

La bandera 53 se usa para controlar la forma en que los objetos se exponen en pantalla cuando las acciones LISTOBJ y LISTAT son usadas. Si esa bandera 53 está seteada a 64 (es decir, tiene el bit 6 seteado, lo cual se obtiene con un comando LET 53 64), entonces todos los objetos serán listados sin líneas entre ellos, formando una sentencia válida; es lo que se llama listado compuesto.

En cuanto a los Mensajes del Sistema implicados en estos procesos tenemos:

MS53 ("nada") - sólo puede ocurrir con la orden LISTAT  
OBJETO Y MS48 (" ")  
OBJETO Y MS47 ("Y") OBJETO MS48 (" ")  
OBJETO Y MS46 (" ") OBJETO MS47 ("Y") OBJETO MS48 (" ")

Además, el Bit 7 de la misma bandera (53), estará seteado (la bandera será mayor de 127, o sea 128) si cualquier objeto se imprime. Esto te permitirá determinar si quieres o no un NEWLINE para resetear los colores.

Una acción del tipo LISTAT debe ser precedida usualmente por un mensaje.

La descripción de **OBJETO** se reconstruye a partir de la descripción completa que se ha puesto en la tabla de texto de objetos.

**Nota Importante:** No se hace ningún chequeo ni para PUTIN ni para TAKEOUT de que objeto utilizado como locno. esté actualmente presente. Esto debes hacerlo tú si es necesario.

## DROPALL

Todos los objetos que sean llevados o puestos encima se crearán en la localidad actual (lo que dará el efecto de que todos los objetos han sido dejados caer) la bandera 1 se pondrá a 0. Este comando se ha incluido para compatibilidad con los antiguos sistemas, (léase QUILL). Es de notar que un DOALL 254 llevará a cabo un verdadero DROP ALL, y además tendrá en cuenta cualquier otra acción especial que se incluya.

Las siguientes seis acciones tratan de versiones automáticas como COGER, DEJAR, LLEVAR PUESTOS, QUITAR DE ENCIMA, PONER DENTRO Y SACAR DE, (GET, DROP, WEAR, REMOVE, PUTIN and TAKEOUT). Son automáticos por el hecho de que en vez de necesitar que se especifique el número del objeto, ellos convertirán el primer Nombre con su Adjetivo en el objeto actualmente indicado o referenciado, buscándolo en la tabla de Objetos Palabras.

La búsqueda se hace por un objeto que esté en una de las siguientes localidades en orden descendente de prioridad. (Mirar las descripciones individuales).

Esta búsqueda con prioridades permite que PAV "sepa" de qué objeto se trata, si más de un objeto tiene el mismo nombre descriptivo (cuando el jugador no haya especificado un Adjetivo), en la localidad actual llevado o puesto encima o en un contenedor (si se trata del comando TAKEOUT).

## AUTOG

Una búsqueda del número del objeto representado por Nombre (Adjetivo) se hace en la tabla de Objetos Palabra según su prioridad de localización que será: aquí, llevado, puesto, etc.

Es más lógico que el jugador esté tratando de coger un objeto que esté en la localidad actual, que uno que lleve o tenga puesto encima, por eso el orden de prioridades.



En la versión castellana se ha tenido el cuidado de modificar esta zona, para que el parser haga una búsqueda a la inversa de las terminaciones de cualquier Verbo, buscando las terminaciones LA, LO, LOS Y LAS, permitiendo así que el jugador pueda decir "COGE LA MANZANA Y COMETELA". Esto facilita mucho la forma de jugar, puesto que no tienes que repetir dos veces manzana.

El parser sabrá que todos los Nombres que tengan un valor menor de 50 los utilizará el escritor para nombres propios, y por lo tanto no se verán afectados por esta regla.

Por ejemplo, en la frase "COGE LA MANZANA DE EVA Y COMETELA", el parser sabe que se refiere a la manzana si está situada con un número mayor de 50 y Eva con un número menor de 50.

La mínima unidad que forma una frase es un Verbo (o un Verbo de conversión, que es un Nombre que tenga un valor de palabra menor de 20, en cuyo caso, si no se encuentra Verbo en esa frase, ese Nombre será convertido en Verbo, es el ejemplo de: NORTE, SUR, etc.).

En caso de que el Verbo sea omitido en una frase, entonces la SL asumirá que se está utilizando un Verbo previamente usado. Por ejemplo: "COGE LA ESPADA Y EL ESCUDO" trabajará perfectamente, porque el parser sabe que lo que estás intentando hacer con el escudo es cogerlo.

Si hay dos o más objetos a los que se refieren las terminaciones "COGELA", "COGEO", etc., se asume por el parser que te refieres siempre al primer nombre. Es decir, funcionará siempre con el primer nombre. Por ejemplo, si dices: "COGE LA ESPADA Y LA DAGA Y TIRALA", como el primer nombre es espada lo que tirarás será la espada.

Examinemos una frase muy larga:

"COGE TODO. ABRE LA PUERTA Y DIRIGETE AL SUR, LUEGO COGE LA MALETA Y MIRA DENTRO DE ELLA".

Si no, el Mensaje del Sistema número 8 "No puedo hacer eso" se imprime. No es un objeto válido, pero sí existe en el juego. De cualquier modo, las acciones NEWTEXT y DONE se ejecutan.

#### AUTOR

Una búsqueda por el número del objeto representado por el Nombre (Adjetivo) 1 se hace en la tabla de Objetos Palabra con una prioridad de: puesto encima, llevado, aquí.

El jugador, lo más probable es que esté tratando de quitarse (REMOVE) un objeto que lleve puesto, que uno que lleve en sus manos o que este aquí.

Si un objeto se encuentra, su número se pasa a la acción REMOVE. Si hay un objeto en existencia en cualquier parte del juego o si el Nombre no estaba en el Vocabulario, entonces el Mensaje del Sistema 23 "No llevo uno de esos" se imprime. Si no, el Mensaje del Sistema número 8 "No puedo hacer eso" se imprime (no es un objeto válido pero existe en el juego). De cualquier forma, las acciones NEWTEXT y DONE se ejecutan.

#### AUTOP LOGNO.

Una búsqueda del número del objeto representado por el Nombre (Adjetivo) 1 se hace en la tabla de Objetos Palabras en el siguiente orden de prioridad: llevado, puesto, aquí.

Es más factible que el jugador esté intentando poner un objeto que lleva en sus manos dentro de otro, que uno que lleva puesto encima o está aquí.

Si el objeto se encuentra, su número se pasa a la acción PUTIN. Si hay un objeto en existencia en cualquier parte del juego o si el Nombre no está en el Vocabulario, entonces el Mensaje del Sistema 28 "No tengo uno de esos" se imprime.

Si no, el Mensaje del Sistema número 8 "No puedo hacer eso" se imprime. No es un objeto válido, pero existe en el juego. De todos modos, las acciones NEWTEXT y DONE son ejecutadas.

## EXPLICACION DEL DIAGRAMA.

Primero se dibuja la parte de fuera del área A y del área B.

El área A se rellena entonces usando una TINTA del primer color.

El área B se rellena usando una TINTA del segundo color.

Entonces se usa el comando BLOCK en el rectángulo indicado por (C) teniendo cuidado de que el PAPEL que se use sea el del segundo color y que la TINTA sea igual a 8, esto deja el área A y el área B sin ningún cambio.

Y por este método el coloreado del PAPEL hace que la confluencia parezca continua.

## COPYOF objno. flagno.

La posición del objeto al que se refiere objno. se copia en el flagno. de la bandera. Esto puede ser usado para examinar la localización de un objeto en comparación con otro valor de la bandera.

Por ejemplo: [COPYOF 1 11 SAME 11 38] puede ser usado para chequear que el objeto 1 esté en la misma localidad del jugador, aunque ISAT 1 255 sería mucho mejor.

## COPYFO flagno. objno.

La posición del objeto al que se refiere objno. se pone como contenido del flagno. de la bandera. Una intención de copiar de una bandera que contenga 255 resultará en un error de "Argumento Inválido". El poner un objeto en una localidad inválida será aceptada puesto que no acarrea ningún peligro para las operaciones de PAW. Esto debes de tenerlo en cuenta (y no hacerlo) tú.

## WHATO

Es un comando muy importante. Una búsqueda para el número del objeto representado por el Nombre(Adjetivo)1 se hace en la tabla de Objetos Palabras en la siguiente prioridad: llevado, puesto, aquí.

Esto es, porque se supone que cualquier uso del comando WHATO se referirá más que todo a objetos llevados que a cualquier objeto que se lleve puesto o esté aquí.

Si un objeto se encuentra su número se pone en la bandera 51, lo mismo que con cualquier parámetro del objeto al cual se refiera actualmente en las banderas 54 a 57.

Esto te permite crear otras acciones de tipo automático (en el manual de introducción dabamos el ejemplo de esto dejando caer objetos desde el árbol).



El usuario debe decidir dónde usar B para comenzar una nueva página, pero sugerimos que uses antes el compresor para ganar toda la memoria posible antes de comenzar otra página, porque luego no se pueden insertar mensajes extra en una página cuando ya se ha comenzado a insertar en otra página, solamente se pueden corregir las entradas ya existentes.

La base de datos inicial, que viene con PAW, tiene la localidad 0 en la página 0 y hay que decidir pronto si la localidad 1 debe ponerse en una nueva página. Sobre todo teniendo en cuenta que todas las otras tablas solamente pueden utilizar la página 0.

Nosotros recomendamos que todos los mensajes y las localidades se pongan en la página 1 y superiores, puesto que la página 0 pronto se llenará de Vocabulario, etc.

Lo mismo ocurre con la tabla de Textos para Mensajes. I insertará siempre en la página más alta seleccionada. B comienza otra página, sólo si ésta ha sido inicializada poniendo una localidad en ella previamente.

Si deseas que tu juego sirva también para un Spectrum de 48K, entonces no debes de usar ninguna otra página excepto la número 0.

## 2. Las Fronteras de Color

Probablemente uno de los aspectos más difíciles de los gráficos en el Spectrum es el problema de los atributos de color.

Esencialmente, sólo se pueden utilizar dos colores en cualquier grupo de pixels de 8x8. Estas áreas se muestran claramente mediante la opción RETICULA en el Editor Gráfico.

Los dos colores son el PAPEL y la TINTA, lo que significa los colores de los pixels apagados y los pixels encendidos respectivamente. Ambos PAPEL y TINTA pueden tener hasta ocho valores para cada grupo de pixels de 8x8.

## ADD flagno1 flagno2

Se le añade a la bandera contenida en flagno2 el contenido de la bandera de flagno1. Si el resultado excede de 255 la bandera se pone a 255.

## SUB flagno1 flagno2

Se le resta al contenido de la bandera en flagno1 el contenido de la bandera flagno2. Si el resultado es negativo la bandera se pone a 0.

## COPYFF flagno1 flagno2

El contenido de la bandera flagno1 se copia en la bandera flagno2.

## RANDOM flagno.

La bandera cuyo número especifique flagno., se pone a un número que es resultado de una secuencia pseudo aleatoria de 1 a 100. Esto puede ser útil para permitir que se tomen decisiones aleatorias de una manera más sencilla que con la condición CHANCE.

## MOVE flagno.

Esta es una potente acción para la manipulación de los personajes pseudo inteligentes (PSI). Trabaja permitiendo que el Verbo actual de la sentencia lógica sea usado para buscar en la tabla de conexiones en busca de la localidad que se dé en la bandera cuyo flagno. se especifique.

Si el Verbo se encuentra, entonces el flagno. especificado cambia a ser el número de localidad asociado con él, y el próximo contacto se considera.

Si el Verbo no es encontrado, o el número de localidad original no era válido, entonces PAW se ocupa de la siguiente entrada en la tabla (si está presente).

De esta forma se podría considerar que PAW ejecuta las siguientes entradas imaginarias a su salida de Respuestas si no ha ejecutado ninguna acción:



En general, es posible usar cualquier Interface de impresora que use el canal 3 correctamente.

Si todo falla, y eres bueno en código máquina, puedes escribir un driver específico de impresora. PAW imprime una dirección cuando carga, (llamada: PRTADD).

Esto nos da las direcciones de los vectores dentro de PAW para el sistema de impresoras, definido de la siguiente forma (el carácter está en A):

```
PRTADD    DEFV PRTRET    ; Dirección del driver de impresora.  
COPYEC    JP    DOCOPY    ; Salto a la rutina de COPY de pantalla.  
PRTRET    RET            ; Pon aquí el driver...
```

Hay espacio libre hasta 29504 para expandir.

Debes grabar tu código de driver desde la dirección correcta, o poniéndole un número de fichero que tenga una "A" en la posición 10.

Junto con él debes grabar un fichero sin bytes con el mismo nombre pero terminado en "B".

Hay que redireccionar entonces el canal número 3, (hacia cualquier otra parte), para hacer creer a PAW que hay una impresora no térmica conectada.

Entonces debe usarse la opción "carga base de datos" del menú principal para cargar tu driver (PAW cree que es una base de datos y cargará ambos ficheros, uno de los cuales será tu driver).

No se hace ninguna prueba para chequear que el jugador ya no lleve con anterioridad un peso mayor que el permitido, porque GET y los demás, (que si tienen en cuenta estos valores), seguirán todavía trabajando correctamente y prevendrán que el jugador lleve más objetos, aún si has seteado el valor a un número más bajo que el número de objetos que ya lleva en este momento.

Hay siete acciones que se usan para la manipulación de banderas, el modo de pantalla, el tipo de formato, el INPUT, etc.:

#### MODE modo opción

Hay cinco modos de pantalla, todos ellos controlados por la bandera 40 y elegidos usando la acción MODE de la siguiente forma:

##### MODO 0

Si un gráfico (diferente de una subrutina) está presente en la base de datos, la pantalla se limpia a los colores por defecto, y el dibujo se pinta. Cuando se pulsa una tecla, los colores originales son restaurados y la localidad es descrita.

##### MODO 1

Sólo texto. Scroll de pantalla entera. El texto va saliendo en la pantalla de forma continua. CLS no es activo durante la descripción.

##### MODO 2

Si un gráfico (diferente de una subrutina) está presente en la base de datos, la pantalla se limpia a los colores por defecto y el dibujo se imprime (como en el Modo 0). Luego, los colores originales son restaurados, y el área que está debajo de la línea dada por la bandera 41 se limpia y el texto se imprime a partir de esa misma línea.

##### MODO 3

Es como el Modo 2, pero el gráfico no hace Scroll.

También puede ocurrir cuando se intenta comenzar una nueva página de RAM en un Spectrum de 48K, o cuando se intenta cargar un juego de 128K en un Spectrum de 48K, o si no se ha dejado suficiente espacio para trabajo, (por ejemplo, si se ha escrito un programa muy grande para ser usado con el comando EXTERN).

#### Invalid Argument

Este error se imprimirá cuando un contacto descubre un valor ilegal. Por ejemplo: ocurre si se intenta cargar la bandera 38 (la que indica la localidad) a una localidad ilegal.

También se puede generar si se intenta poner algo dentro (PUTIN/TAKEOUT) de un objeto que no es un contenedor, o también si se intenta poner un objeto en la localidad 255 (sólo admite COPYFO). (Vease el punto 2 mencionado anteriormente para la información de diagnóstico).

Nota 1. Durante un Input, el Spectrum emitirá un RASP si la pantalla y/o el buffer de input está lleno.

Nota 2. Si una entrada ya está presente en la base de datos y es muy grande para que quepa en una sola pantalla, el Spectrum emitirá un RASP cuando la entrada se sitúe en la parte inferior de la pantalla para corregirla (amending).

Nota 3. Si una entrada anormalmente larga se inserta en la tabla de conexiones usando abreviaciones. Por ejemplo: N 1 - 0 6 - S 4, etc., y las abreviaciones son después borradas del Vocabulario. Entonces nos encontraremos con que quedan las palabras normales: NORTE, OESTE, SUR, etc., las cuales pueden ser muy largas para el buffer de input. Si esto sucede, un mensaje de "out of memory" se producirá. El remedio es reinsertar las abreviaciones en el Vocabulario.

0 - Opción normal, cualquier gráfico que esté presente en la localidad es dibujado cuando el jugador entra por primera vez en esa localidad. A partir de aquí solamente serán dibujados si los gráficos se ponen artificialmente o se pide una dibujada temporal (ver más adelante).

1 - Apagado, cualquier gráfico es ignorado por PAW aunque se pida una redescrípción.

2 - Encendido, cualquier gráfico que esté presente en la localidad se dibuja cada vez que el jugador entre en esa localidad.

Hay una opción 3, (sería la opción 2 + 1) pero la opción 1 APAGADO tendría prioridad siempre, por lo cual es redundante.

El dibujo actual para una localidad puede ser redibujado añadiéndole 128 a la bandera 29, ésta es una dibujada temporal, que puede ser usada cuando el jugador quiera que aparezca el dibujo en el Modo Normal. Por ejemplo, habría que poner en la tabla de Respuestas, como respuesta a su petición: PLUS 29 128 DESC.

#### PROMPT sysno.

Hace que el mensaje del sistema que especifica sysno. aparezca cada vez que PAW obtenga una línea de mando por el jugador. Sería la respuesta del ordenador. Si se deja sin ningún valor, el valor de 0 (por defecto) hará que PAW seleccione uno de los mensajes del sistema 1, 2, 3, 4 o 5 en una proporción de: 30:30:30:10 respectivamente. Hay que hacer notar que esto no afecta a los mensajes que son impresos por medio de los contactos END o QUIT.

#### TIME duración opción

Permite que los INPUT aparezcan después de un tiempo cuya duración se especifica en intervalos de 1,28 segundos.

Por ejemplo, la tabla de Procesos 2 se buscará otra vez si el jugador no tecléa nada durante un tiempo especificado.

Opción, permite que esto también pueda ocurrir durante los mensajes de "PULSAR CUALQUIER TECLA" y "Más....".



## MENSAJE DE ERROR DEL EDITOR Y SU SIGNIFICADO

Debido a la complejidad de PAW, muchas órdenes pueden generar errores.

Cuando un error es descubierto, un mensaje que lo describe se imprime en la parte inferior de la pantalla. PAW, entonces espera a que se pulse una tecla antes de retornar directamente al menú en el cual ocurrió el error.

Hay dos excepciones a esto:

1/ Después de un error mientras se está editando una cadena de dibujo. En este caso, el puntero de la cadena se posiciona justamente antes del comando que causó el error. (Esto significa que un comando del tipo SIGUIENTE, causará el error de nuevo). Si eres incapaz de corregir el problema, entonces debes usar la opción BORRAR SIGUIENTE, para borrar el comando erróneo. Esto puede dejar otros comandos sin dibujar (lo cual puede causar otro error).

2/ Mientras se está jugando la aventura. En este caso, como el Editor de PAW no está presente, se hace un salto hacia la inicialización de un nuevo juego, después de reportar el error. De otra forma, si PAW está presente, una línea de diagnóstico se imprime en la parte inferior de la pantalla. Esta línea muestra la tabla de Procesos, Verbo y Nombre de la entrada, y el contacto en el cual ocurrió el error. A continuación se da la oportunidad de ver un diagnóstico de bandera, lo que ayuda a confirmar dónde ocurrió el error. Cuando ENTER se teclée, el reporte normal de errores se imprime y se hace una vuelta al menú principal de PAW.

Cuando uses tus propios programas junto con el comando EXTERN, cualquiera de los errores típicos del BASIC puede ocurrir. Hay que hacer notar, sin embargo, que la línea y el número donde ocurrió el error no se imprime, por lo tanto hay que probar el programa a fondo antes de que la aventura esté presente.

4 - Causará la reimpresión de cualquier texto que haya sido puesto cuando el INPUT se imprime después de un tiempo muerto. Lo mismo que el anterior, se debe usar principalmente con el número 1 donde puede ocurrir un tiempo muerto entre teclados.

Por defecto PAW asume el INPUT 0 (no opciones).

## PROTECT

Se usa casi siempre en conjunción con el Mode 4. Esta acción lo que hace es que pone la línea imprimirse como tope de la pantalla para el scroll de texto.

Debe ser usada solamente dentro del Proceso 1 (lo puedes poner en cualquier otro sitio, pero no tiene ningún efecto) y después de cualquier texto que se quiera proteger una vez haya sido imprimido. Suele ser usualmente antes o después de una acción LISTOB, puesto que los objetos son usualmente lo último que se imprime.

Puede usarse en otros modos, por ejemplo, para permitir que un gráfico y una parte de una descripción se protejan dejando que el resto del texto haga un scroll por debajo.

Hay tres acciones que imprimen valores que llevan las banderas en la pantalla:

## PRINT flagno.

El contenido en decimal de la bandera a que se refiere flagno. se imprime con los colores actuales sin dejar espacios ni antes ni después. Esta es una acción muy eficaz.

Por ejemplo, si la bandera 100 contiene el número de monedas que lleva el jugador, entonces una entrada en la tabla de Procesos del tipo MES 10 "Tienes (espacio)" y ahora [ PRINT 100 MESSAGE 11 ] "(espacio) monedas de oro.", se usaría para que se imprimiera el número de monedas que tiene.



Ojo Notas: a) Escala solamente afecta a algunos comandos, y son: MOVIDA RELATIVA, LINEA, FILL o RELLENO, SHADE o SOMBRADO. No son afectados ni relocalizados: MOVIDA ABSOLUTA, PLOT, BLOCK y TEXTO, por ello no deben ser usados estos comandos en subrutinas (aunque con la práctica se pueden llegar a hacer funcionar algunas veces).

b) Sólo se pueden anidar llamadas de subrutinas hasta un nivel de diez (anidado quiere decir llamar una subrutina desde dentro de otra subrutina).

c) Escala no afecta al comando GOSUB. Por ejemplo: si GOSUB es usado en una subrutina la cadena dibujada será de tamaño fijo y no en escala.

d) El llamar a la misma rutina con la cual se está trabajando causará un error de "Límite Alcanzado". El mismo error se dará cuando se ha alcanzado el nivel de 10 subrutinas anidadas.

#### 4) ORDENES DE EDICION

→ (CAPS SHIFT + 8) COMIENZO Lleva el puntero al inicio de la cadena.

↓ (CAPS SHIFT + 6) SIGUIENTE Ejecuta el siguiente comando de la cadena, si no lo hay será ignorada

↑ (CAPS SHIFT + 7) PREVIO Mueve el puntero hacia atrás una orden y pone al día la pantalla.

BORRAR (CAPS SHIFT + 0 en 48K) Borra el comando previo de la cadena y pone al día la pantalla.

BORRAR SIGUIENTE (GRAFICS) (CAPS SHIFT + 9 en 48K) Borra el siguiente comando, si lo hay.

#### PICTURE LOGNO.

Dibuja los gráficos definidos por LOGNO. sin importar que sea una sub-rutina o un dibujo principal.

Nota: no se hace un clear de la pantalla y de los colores como cuando se describe una localidad.

Puede ser usado para añadir (o para quitar, si se usa PRINTAT y MES para quitar con espacios) partes de un dibujo.

Hay que tener en cuenta que el dibujo será dibujado a partir del último punto usado por el dibujo que se pintó con anterioridad.

#### PAPER n

Donde n va de 0 a 9. Cambia los colores de fondo del papel.

#### INK n

Donde n va de 0 a 9. Cambia los colores de fondo de la tinta.

#### BORDER n

Donde n va de 0 a 7. Cambia los colores del borde de la pantalla.

#### CHARSET valor

Cambia el set de caracteres que existe usando el valor que se dé si es válido. Si no es válido no se ejecuta ninguna acción.

Los siguientes tres controles de pantalla permiten que una sección de texto sea impresa en la pantalla separada o fuera de la actual posición de impresión. Puede ser usado, por ejemplo, para crear una línea de información en la parte superior del dibujo actual, o para crear una sección de texto en una pequeña ventana dentro de un dibujo, etc, etc.

La trama usada para sombrear viene determinada de la siguiente forma:

a) Se te preguntan 2 números de trama desde el 0 al 15. Si solamente quieres usar una, entonces debes repetir el mismo número para ambas tramas.

b) Las 2 tramas especificadas sufren entonces un proceso de OR (se colocan una encima de la otra).

c) Si el INVERSE estaba encendido (ON), entonces la trama resultante será invertida, (es decir, se cambiarán todos los pixels que estén seteados a pixels reseteados).

Nota 1. El sombreado primero irá en una dirección hacia abajo en la pantalla, y luego de nuevo en dirección hacia arriba. Para una mayor velocidad cuando se está sombreando hacia abajo, no volver nunca hacia arriba hasta que se llegue al final.

Cualquier área sin sombrear debe de hacerse otra vez por separado. Esto se puede evitar escogiendo cuidadosamente el punto inicial.

Nota 2. Si el área que va a ser sombreada es muy compleja (en el sentido de que ya tiene muchos pixels seteados en ella), entonces no se podrá sombrear. Esto se ha tenido que hacer así para que el programa pueda "detectar" si el área elegida ya ha sido previamente sombreada o no. Por eso un área solamente se puede sombrear una sola vez.

No debes sombrear un área y luego intentar llenar el fondo con el comando fill, para ello se debe usar la opción INVERSE.

T = TEXTO (4 Bytes) Si se han insertado sets de caracteres en la base de datos, entonces se pide un número de set. Luego se pide un código de carácter comprendido entre 32 a 162. Este carácter será situado en la pantalla en el cuadrado de caracteres (que se mostrará con el comando GRID), en el cual esté situado la punta de la línea.

## INVEN

El Mensaje del Sistema número 9 "Tengo: \_" se imprime. Si no hay ningún objeto que lleses o que tengas puesto se imprime el mensaje del sistema número 11 "Nada en absoluto".

De otra forma el texto que define cada objeto que se lleva o se tiene puesto encima se imprime en una línea separada.

Si un objeto se lleva puesto, su texto irá seguido del Mensaje del Sistema 10 "Encima" o "Puesto", alineado del lado derecho de la pantalla si no cabe en la misma línea. La acción DONE es ejecutada luego.

Dos acciones hacen que PAW salga completamente de la tabla de Respuestas o de Procesos, son:

## DESC

Cancelará cualquier bucle DOALL, cualquier llamada a sub-proceso y hará un salto para describir la localidad actual.

## END

El Mensaje del Sistema 13 "¿Quieres jugar otra vez?" se imprime y se llama a la rutina de INPUT. Cualquier bucle DOALL y cualquier llamada de sub-procesos son canceladas. Si la respuesta no empieza con el primer carácter que lleve el Mensaje del Sistema 31, se hace un salto a inicialización. Si no, se hace un salto al editor si está presente, o un NEW de BASIC si no está presente.

Tres acciones de salida de tabla:

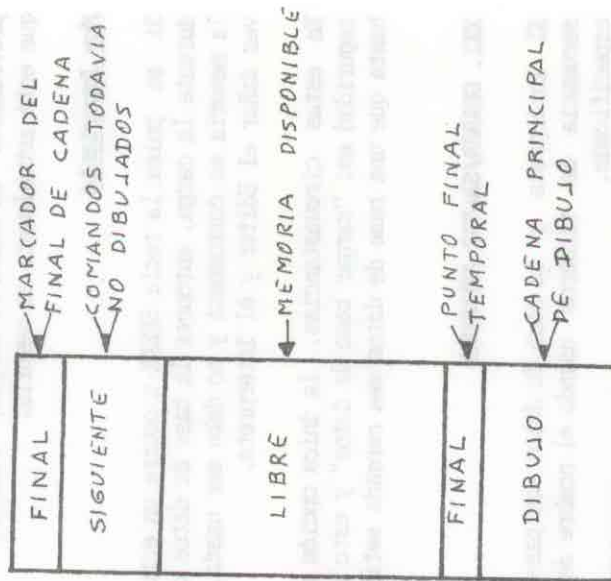
## DONE

Esta acción salta al final de la tabla de Procesos e indica a PAW que ya se ha ejecutado una acción. Significa que no se consideran más entradas. Una vuelta se hará a la tabla de Procesos previa o a el punto de inicio de cualquier bucle DOALL que esté activo.



Descripción detallada del Editor Gráfico

Esta sección de PAW permite llevar a cabo varias operaciones en forma de una cadena de dibujo para cada localidad. Al editar, la cadena se sitúa en la memoria de la siguiente forma:



CARACTERISTICAS

Se usa una línea "elástica".

El "punto" es el punto base de esa línea y muestra el último punto al que se ha movido.

El punto final de la línea elástica muestra la próxima posición del "punto" o el lugar de comienzo de un Fill/Sombreado, etc.

RAMSAVE

Es una forma similar de SAVE, pero que graba toda la información relevante del juego no a cinta, sino a un buffer en la memoria del ordenador. Este buffer, por supuesto, es volátil y se destruye cuando se apaga el ordenador. Esto debe de hacerse saber al jugador. Además, también se limpiará el buffer cuando vuelves a la sección de editor del PAW. Te aviso esto en caso de que cambies el diseño de tu juego.

RAMLOAD flagno.

Esta acción es la contraria de RAMSAVE y permite que el buffer que antes se salvó sea restaurado. El parámetro dado por flagno. especifica la última bandera que debe ser cargada y se usa para mantener valores por encima de un restore. Por ejemplo una entrada de:

```

RAMLO _ COPYFF 30 255 (30 = score)
        RAMLOAD 254      (carga sin cambiar hasta bandera 254)
        COPYFF 255 30    (haz la 30 = 25 y luego describe)
        DESC
    
```

podría ser usada para mantener la puntuación actual y evitar que el jugador pueda usar el RAMSAVE/LOAD como una opción muy fácil de sacar un 100%. Como verás, lo que hace es cargar la puntuación en una bandera y luego devolverle a esa bandera la puntuación.

**Nota Importante:** a diferencia de SAVE y de LOAD las acciones que van a la RAM si permiten que el próximo contacto se ejecute. Por ello, deben de ir normalmente seguidas por una acción DESC para que el juego se restaure en una posición idéntica.

Estas acciones pueden ser usadas para hacer el comando OOPS que se está poniendo de moda en otros sistemas. Se trata de poder volver a la jugada anterior si has metido la pata. Se hace creando una entrada en la tabla de Procesos 2 (o en la tabla de Respuestas) que haga un SAVE a la RAM automático cada vez que el jugador entre su movida.

Nota 1. No se puede corregir un carácter de un set, ni tampoco cargar un set, hasta que éste haya sido insertado.

Nota 2. Las bases de datos para cualquier otro set (excepto 0), son de tipo estándar de 768 bytes. Pueden ser cargadas y definidas desde cualquier otro diseñador de caracteres, mientras el nombre de la base de datos termine con el símbolo "C" en la posición 10.

#### XVII. COMPRESOR DE TEXTO

La pregunta ¿Comprimir base de datos (S/N)? aparece en pantalla. Si la respuesta no empieza con "S" se asume que es negativa y causa una vuelta al menú principal.

De otra forma un diccionario de 222 bytes se inserta en la base de datos.

Se busca entonces en la base de datos por 90 agrupaciones de letras de las más comunes y estas son reemplazadas con un token de un solo byte de los usados en las posiciones 165 a 255.

Un contador de pantalla muestra el número de tokens que aún quedan.

Este método de compresión reduce cualquier texto del juego en aproximadamente un 40%. Esto en un juego de texto solo de 128K, nos dará un equivalente de 160K de memoria libre.

Nota 1. Cuando se hace una edición de texto después de una compresión, el cursor saltará a veces 2, 3, 4 o 5 caracteres debido a la tokenización. Las correcciones deben de volver a ser retécledas completamente de nuevo, puesto que serán recomprimadas la próxima vez que el compresor se use.

#### XVIII. GRABAR/SALVAR BASE DE DATOS

El área de la base de datos se pasa a cinta como una secuencia de ficheros, dos por página. La décima posición del nombre del fichero se marcará con las letras de la A a la L secuencialmente.

- - MESSAGE x ;como arriba u otro mensaje diferente

También hay algunas otras formas de hacerlo:

DECIR nombre SAME pos 38 ;¿Están ellos ahí?  
PROCESS Y ;Decodifica el discurso  
DONE ;Puesto que la SL se ha destruido  
hay que hacer siempre un DONE

DECIR nombre MESSAGE z ;"Ellos no están aquí"  
DONE

#### NEWTEXT

Fuerza que se pierda cualquier frase que permanezca en la línea actual de INPUT. Se debe usar para prevenir que el jugador continúe con un INPUT dañado, sin tener un nuevo INPUT, si algo sucede durante una situación.

Por ejemplo, la acción COGER (GET) ya de por sí lleva un NEWTEXT si por alguna razón falla en coger el objeto que se requiere. Ello se usa para prevenir el desastre en una frase como:

COGE LA ESPADA Y MATA A MANOLO CON ELLA

por si se lleva a cabo un ataque a Manolo sin tener la espada, lo cual es muy peligroso.

Una acción para el sonido:

BEEP duración tono

Ambos, la duración y el tono, van desde 0 a 255. La duración se hace en unidades de centésimas de segundo.

El valor del tono se debe obtener tomando el valor del comando BEEP en BASIC, añadiéndole 60 y luego dividiéndolo por 2.



Inicialmente, sólo estará presente un set de caracteres: el que se contiene dentro de la ROM del Spectrum. Este set también contendrá las 16 tramas de sombreados (desde el código 0 hasta el 15) y los 19 gráficos definidos por el usuario (código 144 al 162) y se denomina set 0.

**Nota:** En la versión castellana se ha definido otro set diferente al de la ROM del spectrum por considerarlo mucho más adecuado.

#### Insertar I

Si aún no se ha alcanzado el máximo número de sets posibles, con esta opción se localiza el siguiente set disponible.

Una entrada de 768 bytes se inserta en la tabla de caracteres y una copia del diseño para los 96 caracteres (códigos 32 al 127) en la ROM se copia en esa entrada. De otra forma ocurrirá un error de "límite alcanzado".

#### Arreglar A un carácter de un set

El carácter especificado del set especificado se presenta en pantalla para su arreglo.

Solamente los caracteres del 0 al 15 y del 144 al 162 pueden ser arreglados desde el set 0, mientras que solamente los caracteres del 32 al 127 pueden ser arreglados en cualquier otro set. Esto es importante.

El Editor crea entonces tres áreas en la pantalla:

- 1/ La zona de enrejado o cuadrículado magnificada (x8) la forma en que está constituido un carácter. Se presenta en forma de cuadraditos amarillos y blancos, donde los bits que estén seteados se mostrarán en negro. También se muestra la posición actual del cursor por medio de un cuadradito rojo parpadeante.

- 4/ Cuando se hace un intento de salida de la tabla actual, si el DOALL todavía está activo (por ejemplo, no ha sido cancelado por una acción), entonces la atención del PAW vuelve de nuevo al bucle DOALL desde el paso 1. La búsqueda continuará desde el siguiente objeto (el que tenga un número más alto).

La consecuencia de este método de búsqueda a través de la tabla de Objetos Palabras es que los objetos que tengan el mismo Nombre y Adjetivo en su descripción, (y en donde el juego sólo sabe a cual objeto se refiere por su presencia en esa localidad) son buscados en un orden ascendente de número de objeto, si no algunos de ellos se perderían.

Si se usa un DOALL para cosas como abrir todo, hay que tener en cuenta el hecho de que, en el funcionamiento de la mayoría de las puertas, se han puesto banderas y que entonces habría que convertir esas banderas en objetos para poderlas incluir en un bucle DOALL.

#### RESET locno+

Esta acción se diseñó para permitir juegos de partes múltiples donde los objetos que no van a ser llevados a la siguiente parte se resetean a sus localidades de inicio.

Todos los objetos que se puedan llevar a la siguiente parte o entre partes deben estar presentes (y con las mismas descripciones) en cada parte. Los demás pueden ser usados de cualquier forma en cada una de las partes.

Cualquier objeto que esté presente en la localidad actual es movido a la localidad designada por el locno. Y la localidad actual pasa entonces a ser la señalada por locno.

Cualquier otro objeto es setead a la localidad que se dió en la tabla de Objetos Inicialmente En.

No tiene ningún efecto en las banderas.

La acción DESC se ejecuta cuando se completa.

Un ejemplo de como se usa este comando se dará en el capítulo donde se habla de los juegos de partes múltiples.

### 3. Imprimir = P (Verbo (Nombre)) o L (Verbo (Nombre))

La impresión puede ser en pantalla usando P, o a la impresora usando L.

La impresión empieza con la primera entrada que tenga los valores de palabra de Verbo y Nombre. Si Verbo y Nombre no se especifican, entonces un número de palabra de valor 0 se supone que es el comienzo. Por lo tanto, apretando P o L solamente sin decir ningún número, comenzarás la impresión siempre al principio de la tabla.

Nota 1. Para borrar una entrada; arréglala para que no quede ninguna condición ni ninguna acción.

Nota 2. Hay un límite de 255 tablas de proceso. Y las llamadas en un bucle anidado pueden hacerse hasta un máximo de 10, después de lo cual ocurre un error.

## OPCIONES EXTRAS (E)

Con esto se seleccionan las otras opciones de que dispone el menú.

## XIII. PROBANDO LA AVENTURA

El mensaje "¿Necesitas diagnóstico?" se imprime, y cualquier contestación que no comience con la letra "S" se tomará como negativa. Si esto sucede, se ejecuta un salto hacia el Intérprete.

Si se requirió o se pidió diagnóstico, entonces mientras se está probando la aventura, y cuando el Intérprete esté pidiendo una rutina de Input, basta con apretar la tecla ENTER para que aparezca el valor de una bandera en la zona inferior de la pantalla.

También se puede teclear el número de cualquier otra bandera para ver su contenido o "=" seguido de un número para hacer que la bandera que esté en pantalla adquiriera ese valor.

Tecleando ENTER otra vez volvemos a la rutina de Input.

## MISCELANEA

Son varias pequeñas tablas y algunos valores sueltos, 50 bytes en total.

## EL SET DE CARACTERES

Esta tabla está vacía hasta que se inserte un set de caracteres usando el Editor de Caracteres. Como es bien sabido, cada set de caracteres usará 768 bytes.

## DICCIONARIO

Esta tabla contiene solamente 1 byte hasta que el compresor se utilice por primera vez sobre la base de datos. Después contendrá 222 bytes de un diccionario de expansión para permitir que los tokens 165 a 255 puedan ser convertidos en las agrupaciones de letras que ellos representan.

## LAS TABLAS DE PROCESOS

Son el corazón de la base de datos puesto que desde ellas se controla el juego.

## LA TABLA DE RESPUESTAS

Cada entrada contiene el valor de palabra del Verbo y del Nombre que va a ser aparejado con la SL, seguido por cualquier número de conductos.

Cuando se juega la aventura, si hay alguna entrada en la tabla que hace juego, o es igual, al Verbo y al Nombre de la SL, entonces el conducto es válido y se ejecuta.

Todos los conductos y lo que hacen, se ha especificado completamente en la descripción del intérprete.

El orden de entrada en la tabla es por valor ascendente de Verbo.



0 - No atributos.

1 - Es un contenedor.

2 - Se puede llevar encima y quitar de encima.

3 - Es un contenedor que se puede poner y quitar de encima.  
(Por ejemplo, una mochila).

2. Imprimir = P o L

Imprimir a la pantalla usando P, o a la impresora usando L. El peso puede estar seguido por una "C" para indicar que es un contenedor o por una "WR" para indicar que se puede llevar encima y quitar (wearable/removeable).

Nota 1. Solamente se puede seleccionar un objeto con atributo de contenedor si hay una localidad que tenga el mismo número. Esto es así, porque PAV trata esta localidad como si fuera "la parte de dentro del objeto contenedor".

Nota 2. Un contenedor de peso 0 no tendrá como peso final la suma de todos los objetos que contenga, si no que su peso será siempre 0. Esto puede ser usado para crear un saco mágico o un transportador por levitación, etc.

## XII. LAS TABLAS DE PROCESOS (incluyendo la de RESPUESTAS)

Solamente se pueden escoger dos opciones del sub-menú principal de Procesos:

1. Comenzar una nueva Tabla = B

Con esto, el siguiente número de proceso que esté libre se adjudica y se crea una tabla nula con este número. El Proceso continúa con una llamada a la opción de Selección de tablas pero con el número de la nueva tabla.

2. Selección de Tabla = S No.

La tabla que se especifique por No. se hace la tabla actual; se selecciona y entonces el sub-menú aparece de nuevo pero con este número de tabla.

## PROCESOS 2

Es buscada por PAV en cada turno. Es decir, después de cada frase sacada del INPUT del jugador, o después de cualquier tiempo muerto durante el INPUT.

El Verbo y el Nombre que se usa en cada entrada en las tablas de Procesos 1 y 2 (y en cualquier sub-proceso llamado desde ellas) no tiene ningún significado, puesto que son ignorados, pero pueden ser usados por el escritor para tener una idea de la función que ejecuta cada entrada dentro de la tabla.

Cada tabla de proceso tiene una cabecera de siete bytes (21 bytes son usados en una base de datos vacía, puesto que tres procesos - incluyendo la tabla de Respuestas- están ya presentes). Cada entrada subsiguiente usa 5 bytes y cada contacto usa 1, 2 o 3 bytes dependiendo del número de parámetros.

**NOTA muy importante:** Si una palabra se borra del vocabulario y no tiene sinónimos, entonces todas las entradas en las tablas de Procesos y de Respuestas que contengan esa misma palabra serán borradas. Cuidado pues con los borrados del vocabulario.

## LA TABLA DE OBJETOS RELACIONADOS CON TEXTO

Esta tabla, que tiene una entrada para cada objeto, contiene el texto que se imprime cuando el objeto es descrito. Cada entrada usa 3 bytes, mas la longitud del texto.

Un objeto es cualquier cosa en una aventura que puede ser manipulado y los objetos son numerados desde el 0 hacia arriba. El objeto 0 es para el intérprete una fuente de luz.

Siempre que se inserta un nuevo objeto, una entrada como no-creado se hace para ese objeto en la tabla de INICIALMENTE EN, otra entrada de " \_ " en la tabla de Palabras Objeto y un peso de una unidad. Sin ser contenedor y sin ser llevable ni quitable se inserta también en la tabla de Objetos Peso.

## VIII. EL TEXTO DE LOS OBJETOS

### 1. Insertar = I

Se escoge el siguiente objeto libre y se hace una entrada en blanco para él en la tabla de objeto texto.

Una entrada de "objeto no-creado" se hace también para ese objeto en la tabla de Lugar Inicial de Objeto.

Una entrada de " \_ \_ " en la tabla de Objeto Palabra.

Se le da el peso de 1 unidad (sin dar más atributos), en la tabla de peso de objetos.

Luego el procesador continúa con una llamada automática a la rutina de arreglo para permitir al usuario corregir la entrada en blanco que ya se ha creado en la tabla de texto de objetos.

### 2. Arreglar = A objno.

El texto que existe para el objeto que especifica objno. se copia en el buffer de INPUT y se muestra en la parte de abajo de la pantalla para su corrección. Cuando se aprieta ENTER, el texto que existe es reemplazado por los contenidos del buffer de INPUT.

### 3. Imprimir = P (objno.) o L (objno.)

La impresión se hace a la pantalla usando P, o a la impresora usando L. La impresión comienza con el texto marcado por objno., o al principio si no se ha especificado ninguno.

Nota 1. El objeto 0 se considera siempre por el Intérprete como una fuente de luz.

Nota 2. Hay un límite de 255 objetos.

## IX. LA TABLA DE LUGAR INICIAL DE OBJETOS

Es donde se especifica la localidad en la cual se sitúa el objeto al comienzo de la aventura.

Una entrada típica sería: SUR 6 ESTE 7 SALIR 6 NORTE 5 lo que significa que SUR o SALIR (o sus sinónimos) producirán un movimiento a la localidad 6. ESTE (o su sinónimo) a la localidad 7 y NORTE (o su sinónimo) a la localidad 5. Cada entrada usa 3 bytes mas 2 bytes por cada par de movimiento.

Nota 1. El par de movimiento contiene el valor de la palabra, no la palabra actual. Esto significa que si el valor de una palabra se borra del vocabulario (borrando todas las palabras que lleven ese número) todos los pares de movimiento que contengan ese valor también serán borrados.

Nota 2. Cuando la aventura completa se está jugando, lo único que producirá movimiento será el Verbo de la SL.

Nota 3. Si se ejecuta un movimiento desde una entrada en la tabla de Respuestas usando la acción GOTO, entonces no se necesita poner ningún par de movimiento en la tabla de conexiones, a menos que esa entrada sea necesaria para un PSI que se puede mover incondicionalmente.

## EL VOCABULARIO

Cada entrada en la tabla usa 7 bytes de memoria y contiene una palabra (o los primeros cinco caracteres de esa palabra si tiene más de cinco letras), un tipo de palabra desde el 0 al 6 y un valor de palabra que va desde el 2 al 254. Los tipos de palabra son:

- 0 - Verbo
- 1 - Adverbio
- 2 - Nombre
- 3 - Adjetivo
- 4 - Preposición
- 5 - Conjunción
- 6 - Pronombre

Las palabras que tienen el mismo valor de palabra son sinónimos. Las entradas se ordenarán según el orden ascendente de valor de palabra.



Nota 1. Hay un límite de 255 mensajes, pero como los Mensajes del Sistema también pueden ser utilizados, ello nos da un total de 510 mensajes (aunque algunos ya están usados).

Nota 2. No se puede comenzar una nueva página de mensajes sin primero inicializar esa página insertando una localidad en ella.

## VII. MENSAJES DEL SISTEMA

### 1. Insertar = I

El siguiente número de mensaje del sistema que esté libre es rellenado con una entrada en blanco en la tabla de mensajes del sistema en la página de RAM 0. Una llamada automática se hace entonces a la rutina de arreglo para permitir al escritor arreglar la entrada en blanco.

### 2. Arreglar = A mesno.

El texto existente para los mensajes del sistema, (o sea mesno.), se copia en el buffer de INPUT y se pone en la parte de abajo de la pantalla para arreglar. Cuando se tecléa ENTER el texto existente es reemplazado por el que contenga el buffer de INPUT.

### 3. Imprimir = P (mesno.) o L (mesno.)

La impresión se hace a la pantalla usando P, o a la impresora usando L. La impresión empieza por el texto especificado con el Mensaje del Sistema mesno., o al principio de los mensajes si mesno. no es especificado.

Nota 1. La descripción del Intérprete nos muestra dónde y cuando se imprimirán esos mensajes. Pueden ser cambiados para segunda persona o primera persona según las preferencias, pero hay que tener cuidado de mantener su significado original.

Nota 2. Los mensajes 30 y 31 no son en realidad mensajes, sino que contienen las respuestas positivas y negativas usadas en los contactos QUIT y END. Por lo tanto, hay que ser muy cuidadosos al cambiarlos, puesto que la acción END es la única y principal forma de volver al Editor desde el Intérprete.

## AREA GRAFICA

El área gráfica de la base de datos crece hacia ABAJO desde la parte superior de la memoria:

## LAS BANDERAS DE LOCALIDAD

Esta tabla tiene una entrada de 1 byte para cada gráfico. El bit 7 especifica si el gráfico puede ser dibujado cuando se alcanza esa localidad (si la localidad no es una sub-rutina). Los bits 0 a 5 describen el PAPEL y la TINTA de comienzo para el dibujo. El bit 6 queda sin usar.

## LA TABLA DE GRAFICOS

Cada entrada en la tabla usa 3 bytes mas la longitud de la cadena. Siempre habrá un número de entradas en la tabla igual al de las localidades creadas para la aventura. La cadena de dibujo esta codificada como una variedad de comandos de diferentes longitudes, lo cual minimiza la cantidad de memoria necesaria para producir el dibujo.

## LOS PUNTEROS

Tiene los principales punteros de la base de datos.

Cuando se tecléa RETURN cualquier espacio libre que aún quede es eliminado. Es importante tener en cuenta que al contrario de cuando se edita texto, aquí la base de datos misma cambia, por lo tanto no se puede abandonar la edición con CAPS SHIFT + 6 (de todos modos no podrías, porque lo que hace es un comando del tipo NEXT).

### 2. Tamaño = S

El número de bytes entre el principio de la base de datos del dibujo y el principio de la siguiente es calculado y se imprime en la pantalla.

### 3. Imprimir, Copiar y Volcado = P picmo., C picmo. y D picmo.

**Picmo.** debe ser especificado.

Si P es seleccionado se pide un nombre de fichero.

El gráfico pedido se dibuja en la pantalla.

Si C fue seleccionado, se llama a la rutina de COPY de la impresora (ver la sección sobre impresoras para detalles).

Si se seleccionó D, un fichero del tipo SCREEN\$ se SALVA a cinta con el número de fichero que se le ha dado (es un bloque de código de pantalla: 16384,6912), este fichero puede entonces ser usado como pantalla de carga para el juego o editado usando otro paquete de utilidad gráfica.

Nota 1. No se puede recargar un fichero de tipo SCREEN\$ en la base de datos.

## V. COLORES POR DEFECTO.

### 1. Arreglar = A picmo. (Papel Tinta)

Una bandera se setea para indicar que el dibujo al que se refiere picmo. es una sub-rutina a menos que el valor del PAPEL y de la TINTA sean especificados, en cuyo caso estos son guardados como los colores por defecto del dibujo. Cualquier localización en la aventura que no requiera un dibujo debe ser Arreglada como una sub-rutina.

Si se especifica el tipo, solamente esa clase de palabras aparecerán en el listado.

Nota 1. Hay que tener cuidado con usar la opción de BORRAR, puesto que puede afectar a la tablas de Procesos, de Conexiones y de Objetos-Palabra. También es una opción que puede llevar bastante tiempo (incluso minutos) si la base de datos es muy larga y hay que borrar muchas entradas que llevan esa misma palabra.

Nota 2. Los Verbos y los Nombres que tengan un valor menor de 14, son palabras de movimiento para el Intérprete, y todos ellos causarían que el Mensaje del Sistema número 7 "No puedo ir en esa dirección" se imprima, en vez del Mensaje número 8 "No puedo hacer eso".

Nota 3. Los valores de palabras del 2 al 254 pueden ser usadas para cualquier tipo de palabra. Las palabras no serán sinónimos si tienen un diferente tipo. Esto permite que más de 250 palabras de cada tipo existan en el Vocabulario.

## II. EL TEXTO DE LAS LOCALIDADES

### 1. Insertar = I

Si ya se ha insertado el número máximo de localidades permitidas, entonces se causará el error "Límite Alcanzado".

Si no, la siguiente localidad libre en la página de RAM que se esté usando (que se muestra en la opción Memoria Libre), se adjudicará y una entrada libre se hace para esta localidad en las banderas de localidad de gráficos, en las tablas de dibujo, en las tablas de conexiones, y en las tablas de texto para localidades.

El proceso continúa entonces con una llamada automática a la rutina de arreglo (Amend) para permitir al escritor corregir el texto vacío que se ha dejado libre.