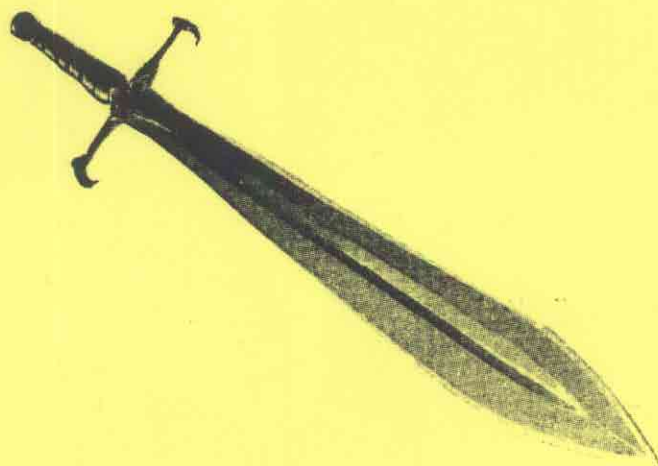


The Professional Adventure Writing System

Introducción



© 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

P A W S

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.

Pero este efecto se cancela la próxima vez que Proceso 1 y Proceso 2 sean ejecutados, porque PAW le resta a la bandera los 128 que le debía.

Esto es para asegurar que los Procesos 1 y 2 actúen con normalidad hasta que se les ordene de una manera específica (con la bandera 58 en 128) que se comporten especialmente.

Suponemos que deducirás hábilmente que si pones 58 a 0, el efecto se anula.

A medida que vayamos introduciendo mejoras recibirás pronta documentación sobre ellas.

INDICE

| | |
|--------------------------|---------|
| Introducción | Pag. 5 |
| Conceptos | Pag. 7 |
| Sección de datos | Pag. 7 |
| Paginación | Pag. 7 |
| Parser | Pag. 9 |
| Menté | Pag. 9 |
| Escribiendo una Aventura | Pag. 12 |
| Enfoque y Guión | Pag. 12 |
| Diseño | Pag. 13 |
| Empezando a teclear | Pag. 16 |
| Localidades | Pag. 16 |
| Colores | Pag. 16 |
| Conexiones | Pag. 20 |
| Probando el juego | Pag. 21 |
| Objetos | Pag. 23 |
| Vocabulario | Pag. 27 |
| Palabras para objetos | Pag. 30 |
| Diagnóstico | Pag. 31 |

DRIVERS DE IMPRESORAS

El Driver de Impresora queda ahora limitado a 48 bytes en la dirección 29587 (PRTADD). Aunque la memoria ha quedado sensiblemente reducida, se ha hecho para poder dar más facilidades extras.

EL PARSE

En ésta versión se han hecho algunos sutiles cambios en la cadena que maneja el PARSE.

Casi todos están encaminados a facilitar e implementar el manejo de los PSI. Pero, aunque no los uses, debes de saber que existen, pues pueden afectar algunas partes del juego.

1-El contacto PARSE, mantiene ahora la "posición actual" dentro de una cadena en la SL que está siendo usada.

Por lo tanto, un segundo contacto PARSE, continuará a partir de esa posición.

En las versiones anteriores, un segundo contacto PARSE, hubiera dado como respuesta la misma SL que el primero.

Por lo tanto la frase ...Dile al Archivero "coge la espada y mata al dragón" es aceptada y ejecutada en cada una de sus acciones.

2-La acción PARSE, no afectará al indicador que marca si la "línea de comando" está vacía o llena, que estaba controlado por la acción NEWTEXT.

Esto significa que una orden del tipo...Decir al Archivero "Coge la llave", ir al Norte....será en esta versión ejecutada correctamente, es decir, primero lo que has dicho al PSI y luego tu acción de irte al Norte.

Introducción

Bienvenido al mundo del escritor de aventuras...

El P.A.W. o Professional Adventure Writer, te ofrece todas las facilidades para producir aventuras gráficas de alta calidad en código máquina.

PAW te proporciona una base estructurada para poder escribir el juego, pero sigue siendo tarea del escritor el tener un guión imaginativo e ideas originales.

Los manuales que vienen con PAW cubren todos los aspectos de su uso. Este manual primero servirá como guía para crear una aventura y recomendamos su lectura detenida y el ejercicio de sus ejemplos antes de intentar hacer un juego propio.

El segundo manual da una información detallada de todo el sistema y debe ser usado como libro de referencia al escribir una aventura.

El equipo de Aventuras A.D. te desea buena suerte y espera tus interesantes producciones.



Parecería que no hay forma de borrar esto sin continuar haciendo PLOTS cada vez más lejos. Pero de hecho, [GRAPH] que es (CAPS SHIFT + 9 en un 48K) borrará el siguiente o próximo comando y podrás borrar también el PLOT de esta manera.

A tener en cuenta es que, al final de una edición, es posible dejarse órdenes por encima del puntero de cadena que sean inútiles. Puedes borrarlas todas usando DELETE NEXT (GRAPH) por un tiempo.

Ahora vuelve al menú principal y selecciona la tabla de Respuestas (los de 48K deberán cargar un OVERLAY). Corrige la entrada * * para que ponga MODE 3 3 LINEA 16 antes de GOTO 2.

La acción MODE 3 selecciona uno de los modos de operar la pantalla. En este caso quedará un área de gráfico fijo, ningún texto la mueve; el segundo 3 le informa a PAW de que no debe cambiar el color del borde y que debe imprimir "CONT.." cuando la pantalla esté llena de texto.

LINEA 16 le dice a PAW dónde va a empezar la primera línea de texto.

Finalmente usamos TEST para probar la aventura (de nuevo los de 48K deben de cargar un OVERLAY) para verla en acción.

El dibujo solamente será pintado la primera vez que visites la localidad, pero no en visitas siguientes. Esto es lo que se llama el modo normal para gráficos. Es posible de todos modos seleccionar ON y OFF, siendo ON la opción para siempre poner los gráficos y OFF nunca poner los gráficos.

Se seleccionan con: PICS NORM

PICS ON

PICS OFF

Esas órdenes no fueron explicadas cuando hablamos de la tabla de Respuestas, pero están todas en el manual técnico. (Que también habrás de leer, je, je).

Debes decidir si permites al jugador cambiar entre estas opciones o no, y forzarle a usar un solo método desde el comienzo.

Conceptos

Es una buena idea introducir algunos conceptos importantes utilizados por PAW.

Secciones de datos

Durante muchos años ha sido de práctica común en los ordenadores que utilizan disco como soporte principal, el partir los programas largos en dos o mas secciones que se cargan desde el disco cuando se van a utilizar. Este sistema no ha sido muy utilizado en cintas por su misma naturaleza como contenedor de datos en forma seriada.

PAW usa una forma muy simple para dejar el máximo de memoria libre, tanto en 48K como en 128K. Utiliza la memoria disponible para ir guardando porciones de datos hasta el momento en que esa memoria va siendo usada y sobrescrita. Los datos entonces se van obteniendo desde cinta según se requieran.

En un 128K no hay necesidad de preocuparse por las secciones de datos hasta que las últimas 16K son usadas (i es decir, hasta que se lleva escrita una aventura de cerca de 92K!).

En 48K las secciones se empezarán a necesitar en el momento de usar los gráficos o al llevar escritas ya 16K de programa.

La muestra en este manual no necesita secciones extra de datos así que su uso se explicará en detalle mas adelante.

Paginación de memoria

El 128K usa un sistema llamado paginación para obtener su memoria extra; el diagrama 1 ayudará a visualizar su composición. En él se vé que la parte superior se puede mover como una diapositiva en un proyector para que cada una de las páginas de 16K se pueda acomodar en un ordenador que solo puede ver 64K cada vez. (ver diagrama No. 1)

INVERSE hace que cualquier PLOT y LINEA reseteen o apaguen los pixels afectados en vez de setearlos o encenderlos, por lo tanto:

MOVIDA RELATIVA 141,95 LINEA 141,107
MOVIDA RELATIVA 146,108 LINEA 146,96

ahora quita el INVERSE con [SS] + [I].

LA CERCA

Vamos a usar una técnica importante de desdibujar el perímetro de un diseño de sombreado (SHADE):

Tinta negra [SS] + [X], [0 ENTER].

PLOT 8,104 LINEA 79,104 LINEA 79,88 LINEA 8,88
PLOT 247,88 LINEA 208,88 LINEA 208,104 LINEA 247,104

Usamos el diseño de SHADE número 7 en 246,102 y 11,102.

Vamos a desdibujar la parte de arriba de la cerca para crear algunas pequeñas puntas: [SS] + [I] (INVERSE).

PLOT 247,104 LINEA 208,104 PLOT 79,104 LINEA 8,104

Luego quita el INVERSE.

Para terminar el efecto:

PLOT 8,101 LINEA 79,101 PLOT 208,101 LINEA 247,101

Vamos a añadir ahora unos matorrales de hierba usando la sub-rutina que mencionamos anteriormente. Esto nos economizará memoria y tiempo. Con ENTER vamos al submenú. Como la localidad 0 es la pantalla del título vamos a usarla en nuestro juego de demostración para que su cadena de dibujo contenga nuestros matorrales de hierba.

Usamos [A 0 ENTER] para corregir el dibujo de la localidad 0. Nos damos cuenta de que el número está seguido por la letra S, que indica que es una sub-rutina (no se pintará).

Parser

To Parse: clasificar una palabra o analizar una frase en sus términos gramaticales.

PAW convierte lo tecleado por el jugador en series de sentencias lógicas (SL) para las cuales ya hay una respuesta definida.

Esto se hace extrayendo frases de la cadena de input, de una en una y dejando que el intérprete las descifre.

Las frases pueden ir separadas por cualquier signo de puntuación y por las conjunciones *y*, *entonces*, o *luego* (o la que se haya definido).

Cuando ya no hay mas frases en la actual cadena de input, se pide otra al jugador.

Una frase debe consistir al menos en un **verbo** (indica acción); opcionalmente dos **nombr**es (objetos o personajes); posiblemente asociados a **adjetivos** (que los describen); de **adverbios** (que modifican al verbo); de una **preposición** (que muestra la relación de un nombre con otra palabra);

y también puede contener una "cadena entre comillas para hablar con otros personajes"

Ment

Del menú principal, mostrado al pulsar cualquier tecla desde la página de título, se puede acceder a cualquier opción del PAW con sólo teclear una sola letra (en MAYUSCULAS) seguido de ENTER.

El menú está dividido en dos partes y la opción E permite cambios entre ellas. La parte inicial muestra todas las funciones necesarias para crear un juego. La segunda (accesible con E), contiene todo lo necesario para Salvar, Cargar, Verificar, Probar el juego, etc. y también el diseñador de caracteres y el compresor de textos (se explicarán mas adelante).

PLOT 206,96 LINEA 206,109 LINEA 162,109 LINEA 162,96

mueve ahora la línea a 163,97 y haz un SHADE con el diseño 15 (el que habías diseñado en la sección sobre el Editor de Caracteres), se hace con [SS] + [S], [15 ENTER 15 ENTER].

PLOT 125,96 LINEA 125,109 LINEA 81,109 LINEA 81,96

mueve la línea ahora a 82,97 y utiliza el diseño de SHADE número 15 otra vez.

Vamos ahora con los palos del Pabellón de Música:

PLOT 103,96 LINEA 103,136 LINEA 106,136 LINEA 106,96

mueve la línea a 104,121 y con [SS] y [F] accedemos al comando FILL: éste llena un área previamente definida completamente con pixels seteados en un forma muy similar a SHADE.

MOVIDA RELATIVA 184,96 LINEA 184,136 LINEA 181,136 LINEA 181,96

mueve la línea a 183,122 y haz otro FILL con [SS] + [F].

Veamos ahora el techo del Pabellón:

MOVIDA RELATIVA 207,143 LINEA 206,140 LINEA 202,136 LINEA 86,136
LINEA 83,139 LINEA 80,143 LINEA 207,143

mueve la línea a 205,141 y haz un SHADE con el diseño 15.

MOVIDA RELATIVA 144,166 LINEA 220,139
MOVIDA RELATIVA 65,139 LINEA 144,166

mueve la línea a 144,164 y haz un FILL.

CREANDO UN EFECTO DE VOLUMEN EN EL PABELLON DE MUSICA:

Selecciona OVER con las teclas [SS] + [O], aparecerá una letra O en la línea de estado para indicar que el OVER está activado.

La bd se salvará en varias partes. PAW salva dos ficheros por página, cambiando la última letra del nombre a A, B, C, etc para cada uno usado.

Con H se verificará la bd salvada comparándola con la que hay en memoria. Si se detecta un error, entonces hay que volver a salvar la bd en una cinta diferente si es necesario.

Con J se pide el nombre de la bd que se desees cargar de nuevo dentro de PAW. *Cualquier base de datos que esté presente dentro de PAW se borrará para dejar paso a la nueva.*

Si se produce un error de carga, se romperá toda el área donde se aloja la bd y la única opción que se debe hacer es continuar con J hasta que la bd se haya cargado satisfactoriamente. Cualquier otra opción producirá daño al programa propio de PAW y se necesitará una recarga completa.

Regla: salva tu base de datos regularmente y en cintas diferentes para tener así un resguardo en caso de producirse el clásico corte de luz u otro accidente cualquiera.

LA ESCALERITA: Mover la línea a 144,78.

Con [SS] + [R] (MOVIDA RELATIVA): Una MOVIDA RELATIVA es la que mueve el punto como PLOT y como la MOVIDA ABSOLUTA, pero a un pixel que está a una distancia fijada del punto actual, en vez de hacerlo a una posición absoluta X, Y. Es usado para mantener juntos los grupos de órdenes que dibujan un solo objeto en un gráfico. La razón de esto la veremos en un momento. Fijemos la línea a 112,78.

[SS] + [R] (MOVIDA RELATIVA) para mover a 111,81. Fijemos la línea a 143,81 y continuamos por los siguientes grupos de coordenadas:

MOVIDA RELATIVA a 144,84 LÍNEA a 112,84
MOVIDA RELATIVA a 111,87 LÍNEA a 143,87
MOVIDA RELATIVA a 144,90 LÍNEA a 112,90
MOVIDA RELATIVA a 111,93 LÍNEA a 143,93

Ahora vamos a examinar las facilidades de edición que podemos usar para corregir errores, aparte del DELETE ya conocido. Como ejemplo, vamos a mover toda la base de ladrillos que acabamos de crear, dos fronteras de caracteres a la derecha. El borrarla con DELETE toda entera para volverla a pintar, sería un poco palizas.

Vamos a explicar cómo guarda PAW la cadena de dibujo. Mientras tú pintas tu obra maestra, PAW va añadiendo cada orden a la cadena. El sitio donde la añade se llama el "puntero de cadena" y de momento el puntero de cadena es el final de la cadena.

Es posible para PAW retroceder a través de las órdenes hasta cualquier punto de la cadena. Para ello se usa el cursor de la derecha (CAPS SHIFT + 8) una sola vez, no hay que preocuparse si desaparece, puesto que el dibujo todavía está ahí. Pero aunque el dibujo está todavía en memoria, PAW sólo lo pinta hasta donde esté el puntero, el cual estará ahora al comienzo de la cadena.

El diagrama 7 te facilitará la visión de como se sitúa la cadena en la memoria.

Puedes adelantar al siguiente comando con cursor abajo (CAPS SHIFT + 6) y puedes retrasar al anterior comando con cursor arriba (CAPS SHIFT + 7). Esto es lo que se llama "Comando Previo", no borra la orden, sólo mueve el puntero una orden hacia atrás.

Diseño del juego

Ahora que la idea está mas clara es importante hacer un boceto del área de juego, en nuestro ejemplo sería algo parecido al diagrama No. 2.

Es de notar que la zona de juego debe estar cerrada de una manera lógica, o el jugador no entenderá por qué no puede ir en una dirección si no hay nada que le impida el paso.

Para el ordenador, una aventura consiste en un determinado número de apartados o localidades que el jugador debe visitar, es ahora el momento de decidir qué zonas del boceto se tomarán como localidades y numerarlas individualmente.

Se debe intentar hacer una escala consistente o lógica (a menos que el juego sea ilógico intencionadamente), evitando que con un solo paso se vaya de una localidad a otra que aparentemente queda a varias millas de distancia.

La localidad 0 debe ser reservada siempre como pantalla de títulos y dejaremos libre también la 1 para darle un uso especial, por lo tanto comenzaremos desde la 2 hacia arriba.

En nuestro ejemplo hemos escogido 7 localidades:

- 2- parada del autobús
- 3- en el jardín
- 4- cerca al banco
- 5- en el pabellón de música
- 6- el estanque
- 7- al lado del árbol
- 8- arriba del árbol

Ahora podemos empezar las descripciones de las localidades. Deben ser lo mas imaginativas posibles, pero en estilo corto e interesante. Hay que procurar mantener una misma forma de verbo todo el tiempo, generalmente primera (yo) o segunda (tu) persona, o pronto el jugador tendrá una seria crisis de identidad. Sea cual sea la forma elegida, debe de estar de acuerdo con los Mensajes del Sistema (ver mas adelante).

5/ A veces, si quedan sitios sin rellenar, se puede hacer poniendo el punto de comienzo dentro de la zona vacía.

6/ Hay 16 diseños que se pueden cambiar usando el Editor de Caracteres. (Ver Editor de Caracteres).

7/ Si se cambia un diseño, este se cambiará en todos los dibujos en los que haya sido usado. Por eso recomendamos usar la mayoría para un uso general y definir unos cuantos para usos muy especiales.

8/ Por supuesto, se pueden mezclar diversas combinaciones de diseño.

Ahora vamos, en nuestro dibujo, con el cielo y con la tierra.

Ponemos el color del papel moviendo la línea a 8,56 y apretando [SS] + [A] para una MOVIDA ABSOLUTA. Esta Movida Absoluta no afecta al pixel como lo hacia PIOT.

Ahora, si tecleamos [SS] + [Y] quitaremos la retícula, pues mientras esté activa NO se ven los colores.

Para poner la hierba de color verde, apretaremos [SS] + [C] (para el papel), y como verde es el color 4 teclearemos [4 ENTER] (fijate como en la línea inferior de la pantalla el número del papel cambia a 4).

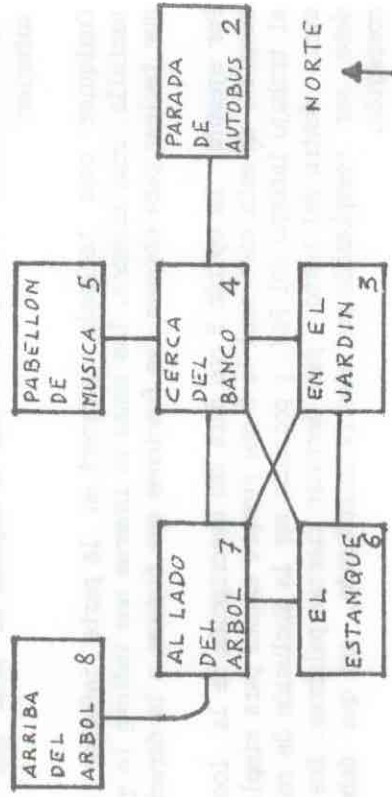
Ahora hay que mover la línea a 246,87 y apretar [SS] + [B] (de BLOCK): esto coloreará el rectángulo del cual la línea forma la diagonal con el actual color para el papel y para la tinta.

El azul del cielo: haz [SS] + [A] o una Movida Absoluta hacia 247,88. Seleccionemos ahora papel azul con [SS] + [C] y [1 ENTER], movámonos a 8,167 y hagamos otro BLOCK con [SS] y [B]. Veremos como el área cuya diagonal estemos señalando se tiñe del color que hayamos elegido.

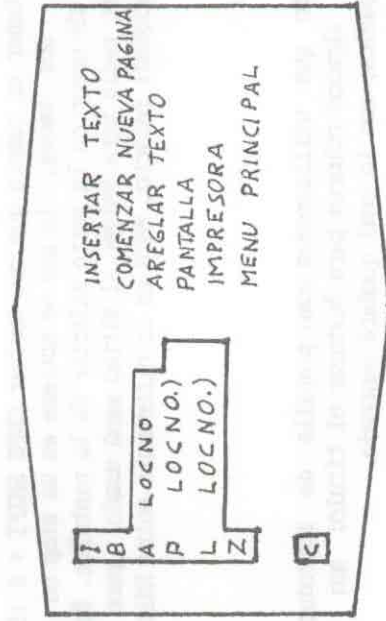
La base del Pabellón: con ladrillos rojos.

Para claridad y ver mejor los posibles movimientos haremos el mapa del diagrama 3.

Diagrama 3



Pantalla de muestra del menú



El movimiento es pixel a pixel, pero se puede acelerar a ocho pixels cada vez si se mantiene apretado CAPS SHIFT al mismo tiempo que las teclas de movimiento o mientras se maneja el Joystick.

La línea que se usa para dibujar es del tipo llamado "elástico", porque se comporta como un tenso elástico mientras se mueve por la pantalla, con ello se consigue una mayor exactitud al posicionarla para el dibujo.

En nuestro juego de ejemplo vamos a usar gráficos a media pantalla, por lo que necesitamos dejar varias líneas libres en la parte inferior de la pantalla para el texto.

Con [SYMBOL SHIFT] más [Y] activamos la Reticula para mostrar las fronteras entre caracteres.

Ahora hay que mover la línea a X=0, Y=47 y luego mover el "punto" al final de la línea con [SYMBOL SHIFT] y [P] (Plot) para que el dibujo comience en ese pixel.

Lo siguiente es pasar la línea a X=255, Y=47 usando [A] sólo una vez, lo que de paso sirve para demostrar que la pantalla no tiene fronteras en los bordes y que al salir por un lado, entrarás por el otro (esta opción, al parecer insignificante, te ahorrará mucho tiempo en tus dibujos).

Ahora pasamos a dibujar la línea, lo que se llama "fijarla". Para ello apretamos [SYMBOL SHIFT] y [L] (De Línea) o usando la opción "fuego" del Joystick.

Si se hace un error, se puede borrar una orden previa con DELETE (CAPS SHIFT y 0 en un 48 K), lo cual te llevará hasta el inicio del dibujo si así lo deseas.

NOTA : Mientras no se especifique lo contrario, todas las órdenes de inserción en la cadena de dibujo (como PLOT o LINE) requieren el uso de SYMBOL SHIFT, por lo cual lo abreviaremos a SS.

NOTA : Todas las coordenadas se darán en forma X,Y. Por ejemplo: 255,47 en vez de continuar con el engorroso X=255, Y=47.

Teclando EXTRA MODE (que es Symbol Shift + Caps Shift en el 48K), los números del 0 al 7 darán el papel o color de fondo. Si nosotros queremos que el título tenga un fondo rojo apretaremos el 2.

Tecléa el título, que en nuestro caso es "EL BILLETE", y ahora debemos volver al color original de papel, así que tecléa EXTRA y 0 para obtener el negro.

Necesitamos poner una línea en blanco entre el título y el texto, pero no podemos tecliar ENTER porque esto termina la edición, por lo tanto, necesitamos UN CODIGO EXTENDIDO DE CONTROL DE PANTALLA (CECP); estos son códigos de 0 a 7 que tienen varios usos dentro del PAM.

CECP 7 da automáticamente un newline (INTRO o nueva línea).

Para usar un CECP hay que hacer una pequeña trampa con el Editor de la siguiente forma: Primero tecléa EXTRA y selecciona el color blanco (7), luego tecléa DELETE (borrar) una sola vez; con ello borrarás el control del papel pero dejara el número 7, por lo tanto, como dijimos antes, el cursor saltará al principio de una nueva línea. Haz esto otra vez para dejar una línea en blanco y tecléa el resto de la introducción. Ten en cuenta que PAM hace un formateo automático del texto.

TEXTO: [Mientras esperaba el autobús una rafaga de viento se llevó mi billete. ¿Puede Vd. ayudarme a encontrarlo?].

Cuando hallas terminado tecléa ENTER para finalizar la edición y luego cualquier tecla para volver al submenú.

Con la opción P se puede ver el texto todavía sin formatear, pues esto sólo se ejecuta mientras se juega el juego.

Con L se imprimirá el texto. Si parece que el ordenador se cuelga tecléa BREAK (CAPS SHIFT + ESPACIO en 48K) para volver al menú, y entonces busca en la guía técnica más información sobre la impresora.

Con I se crea una nueva localidad. No hace falta tecliar un número detras porque PAM automáticamente le asigna el próximo que esté libre.

Como ejemplo para uso de los gráficos, lo que haremos será dibujar el gráfico del Pabellón de Música como se vería desde la localidad 4 (el camino).

Como queremos que PAW dibuje el gráfico cuando visite la localidad, tendremos que asignarle a nuestro gráfico un valor para Papel y para Tinta.

Usaremos un fondo negro (o sea un papel negro) y una tinta amarilla: Tecléa [A 4 0 6 ENTER]. Si ahora vuelves a usar [P] verás que en la entrada para la localidad 4 dirá:

Localidad 4 Papel: 0 Tinta: 6

Si aún no estás contento y quieres cambiarlo, tendrías que usar la opción de corregir (por ejemplo: A 4).

Seleccionemos la opción G del menú.

Los usuarios de 48K se habrán sorprendido al ver que no necesitaban cargar otro overlay, esto es porque Caracteres, Colores por Defecto, y Gráficos están todas en el mismo.

Si usas la opción Arreglar limpiará todos los overlays que PAW tenga en memoria y si deseas usar otra opción, excepto CARGAR/GRABAR o Memoria Libre, necesitarás cargar el overlay que las contiene.

Pero continuemos con los gráficos.

Para cada dibujo del juego hay varias opciones que permitirán: Arreglar, Imprimir en pantalla, Copiar a una impresora, calcular el tamaño de un dibujo o hacer un Volcado (Dump) de la pantalla.

Como nosotros vamos a hacer el dibujo de la localidad 4, hemos de teclear [A 4 ENTER]. La pantalla se limpiará dejando sólo dos líneas de información en la zona baja.

La información proporcionada, de izquierda a derecha es la siguiente:

TEXTO 4: [Estoy en un camino de grava que va de Este a Oeste, muy cercano a un cómodo banco. Hacia el Sur hay un cuidado césped y hacia el Norte hay un pabellón de música].

TEXTO 5: [Estoy en el pabellón de música. Al Sur hay un camino de grava].

TEXTO 6: [El sol crea bellos efectos de luz en la superficie del estanque, mecida por una suave brisa. Hay un camino al Norte que termina en un lloroso sauce. Al Este queda un cuidado césped].

TEXTO 7: [Estoy al lado de un sauce llorón. Al Sur queda el estanque].

TEXTO 8: [Estoy sentado en una rama del sauce, con una visión panorámica del parque lejos al Este, más allá de la verja del parque, puedo ver la parada del autobús].

Usa P para chequear lo que has tecleado y cuando la pantalla se llene de texto aparecerá el mensaje "más..." en la parte inferior de la pantalla. Si entonces pulsas cualquier tecla (excepto BREAK, ESPACIO o N) aparecerá otra pantalla llena de texto y así hasta llegar al final donde aparece el mensaje "PULSE CUALQUIER TECLA".

BREAK, ESPACIO y N causarán un BREAK error y te permiten salir del listado.

Vale. Ahora tenemos varias localidades pero no hay modo de llegar de una a otra.

Eso nos deja dos opciones por ver; la opción B con la cual se inicia una nueva página de memoria en los 128K. De momento no tocarla, porque el juego de práctica cabe sobradamente en la página 0.

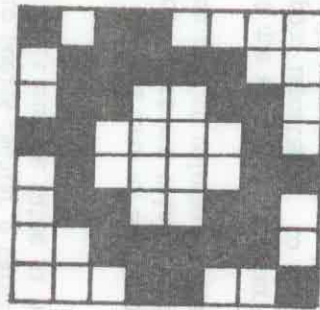
Sobre la opción B, trataremos más extensamente en la guía técnica.

Con la opción Z, como agudamente habrás deducido, volveremos al menú principal.

Usa las teclas del cursor (CAPS SHIFT 5 hasta 8 en un 48K) para mover el cursor rojo parpadeante alrededor del entramado, y la tecla de ESPACIO para cambiar la situación de un pixel. Es decir si el pixel está encendido (negro) lo pondrá apagado; y si está apagado (amarillo o blanco) lo encenderá. Intentalo y verás lo que queremos decir.

El sombreado o el diseño que requerimos es el que muestra el Diagrama 5. Cuando hayas terminado usa [R] para volver a dibujar los dos cuadros donde se muestran los sombreados y verás la nueva forma de entramado. Finalmente pulsa ENTER para terminar la edición.

Diagrama 5



La cara B de la cinta contiene 22 diferentes sets de caracteres, que pueden ser cargados en los sets de caracteres del 1 al 5, después de haberlos insertado, por supuesto.

Después de la inserción de un set, la opción de usar ése set como set principal o set por defecto se ofrece en el menú de Colores de Fondo.

Las conexiones que quedan son las siguientes:

- Localidad 4 (N 5), (E 2), (S 3) (SO 6) (O 7)
- Localidad 5 (S 4), (SO 7)
- Localidad 6 (N 7), (NE 4), (E 3)
- Localidad 7 (ARRIBA 8), (NE 5), (E 4), (SE 3), (S 6)
- Localidad 8 (BAJAR 7)

Ahora corrige las entradas de la localidad 0 para que cualquier movimiento nos lleve a la localidad 2, donde comenzamos el juego (hay una forma mejor de hacer esto, pero necesitamos para ello usar una tabla que todavía desconocemos, y que ya te mostraremos más adelante).

Ahora solamente falta teclear (A 0) ENTER (N 2) ENTER, y ya nos indica que de la localidad 0 al NORTE iremos a la 2.

Comprueba todas estas entradas mirando el mapa hasta que estés seguro de que todo sea correcto. Recordamos que usando la Z se puede volver al menú principal y que desde el menú principal con la opción S se puede grabar en cinta la base de datos actual.

También puedes, con la opción E, examinar el resto del menú que no cabe en la pantalla.

JUGANDO EL JUEGO

Ha llegado el momento de intentar probar nuestro juego. La opción requerida para ello es (T) en el menú principal. Te preguntará si quieres diagnóstico o no, por el momento sólo tecleamos N y ENTER, porque no sabemos todavía para qué sirven los diagnósticos.

Ahora debe de aparecer la pantalla con el título que tecleamos al comienzo. La línea de INPUT se usa para poner las órdenes para que el PAW las interprete y ejecute tus acciones de acuerdo con la información que le hayas dado cuando escribiste el juego.

De momento, solamente le hemos dicho hacia donde nos puede llevar cuando una dirección se teclea. Eso es lo que intentaremos probar.

La opción K del menú principal (los usuarios de 48K tendrán que cargar un overlay en este punto) preguntará si quieres comprimir la base de datos, cualquier tecla que teclées aparte de la S de "si" te mandará de vuelta al menú principal.

El compresor de texto reducirá la cantidad de memoria que se necesita para el texto de tu juego agrupando las letras más comunes en "tokens". Esto puede tomar desde un minuto hasta una hora dependiendo del tamaño de tu juego. En este juego de demostración que estamos haciendo llevará más o menos un minuto y salvará unos 900 bytes.

La única diferencia que encontrarás es que, cuando estás editando un texto que ya existe, el cursor pegará saltos de dos, tres, y a veces cuatro y cinco caracteres de una sola vez. Lo mismo pasará al borrar. Si se hace alguna corrección hay que teclear todas las letras separadas, puesto que ya serán comprimidas la próxima vez que uses el compresor. Es de notar que el compresor usa los tokens normales del Spectrum, lo que producirá agrupaciones de letras, pero no de las palabras claves después de que uses el compresor.

Muy importante es que no uses los tokens, o sea, las palabras clave, si luego vas a comprimir la base de datos, (es decir, no pongas tokens dentro de tu texto si después vas a comprimir). Y para saber lo que son tokens mira el manual de tu ordenador.

OBJETOS

Objeto es cualquier cosa que el jugador puede manipular dentro del juego, por ejemplo: una manzana que puede comer, una llave que puede usar para abrir una puerta, una espada para pelear, etc.

En nuestro juego, que es bastante simple, usaremos los siguientes objetos (no todos ellos tendrán una función en el juego finalizado):

| | |
|----------|-------------------------|
| Objeto 0 | una antorcha encendida. |
| Objeto 1 | una bolsa. |
| Objeto 2 | un emparedado. |
| Objeto 3 | una manzana. |
| Objeto 4 | un billete de autobús. |
| Objeto 5 | una piedra. |
| Objeto 6 | un anorak. |
| Objeto 7 | una antorcha apagada. |

Muy Importante: La antorcha se trata como dos objetos separados. Cambiaremos de uno al otro cuando el jugador la apague o la encienda.

Otro punto a destacar es el hecho de que PAW, al hacer un inventario, muestra la descripción de los objetos eliminando la primera palabra ("una"). Para evitar esto, se deberá insertar al comienzo de cada descripción un código de cambio de color, por ejemplo [EXTRA Y 0] para poner el paper negro. Este truco evita que PAW se coma la primera palabra.

La opción 0 del menú principal sirve, como muy agudamente habrás deducido, para poner las descripciones de los objetos, de una forma muy similar a como hemos puesto las descripciones de las localidades.

Veremos que también existe un objeto 0 si se usa la opción P, para hacer una lista de ellos.

Por lo tanto hay que corregir esta entrada del objeto 0 tecleando (A 0) y usando un color de tinta diferente. Por ejemplo, si queremos usar el color Cyan, hay que pasar al modo extendido o extra en el 128 y luego teclear al mismo tiempo CAPS SHIFT + 5.

EL USO DE LOS OVERLAYS

El usuario de un 128K probablemente no necesita usar paginación todavía, pero puede encontrar beneficioso también este capítulo.

Ya explicamos antes el concepto de las paginaciones. Para poder pasar los gráficos y la comprensión del texto, por ejemplo, los usuarios de 48K tendrán que cargar un overlay o página aparte.

PAW hará casi todo el trabajo para tí. Si seleccionas la opción de menú que necesitas, se te pedirá que confirmes si quieres cargar o no un overlay. Pulsando cualquier tecla que no sea la S o la Y (de yes) te mandará al menú principal. Si continúas, PAW imprimirá el nombre del overlay que está buscando en la pantalla.

Los cinco overlays se encuentran al final del programa principal, que es donde la cinta quedará situada después de cargar el PAW. Si tienes un contador de revoluciones, es importante que lo pongas a cero en este punto, hagas una lectura de donde comienza cada overlay y así puedes ir directamente a cada uno de ellos.

Los cinco overlays están en el orden que mostramos más abajo, y también describiremos las opciones que contienen cada uno:

PAWOVR 1 Lleva el Interpreté, el probador del juego, Grabar y Cargar la aventura ya terminada.

PAWOVR 4 Lleva las tablas de Procesos y Respuestas, el Vocabulario, las Conexiones, y Palabras.

PAWOVR 5 Lleva los Mensajes, Localidades, Peso de Objetos, Objetos-Inicialmente-En y Colores de Fondo.

PAWOVR 2 Lleva un Compresor.

PAWOVR 3 Lleva un Editor de Caracteres y el Editor Gráfico.

Es de notar que el Grabar, Verificar y Cargar una base de datos, junto con la Memoria Libre, están siempre a tu disposición, puesto que son parte del menú principal.

Objeto 1 2; porque la bolsa se encuentra en la parada del autobús.

Objeto 2 254; porque el jugador lleva el emparedado.

Objeto 3 254; y la manzana.

Objeto 4 8; significa que el billete está arriba del árbol.

Objeto 5 3; una piedra que está en la hierba.

Objeto 6 253; porque el jugador lleva puesto encima el anorak.

Objeto 7 254; y lleva consigo la antorcha apagada.

Ahora vamos ha decirle a PAW más cosas acerca de los objetos; por ejemplo, su peso relativo, o si son capaces de contener a otros objetos, o si el jugador puede llevarlos puestos encima.

Volvemos al menú principal y tecleamos la opción X, con ello pasamos al menú de peso de objetos, que también nos permite poner los otros dos atributos (contenedor y llevable encima).

Como siempre, PAW ya ha creado una entrada en esta tabla para todos los objetos que se han insertado previamente. De hecho ya le ha dado a cada objeto el peso de una unidad, no contenedor, y no llevable. Esto lo asigna a cada objeto por defecto (por defecto significa el que PAW le dá, si tu no le das ningún valor).

Necesitamos probablemente corregir el anorak y la bolsa, puesto que el jugador es capaz de llevar puesto encima el anorak (de hecho el jugador está haciendo esto desde el principio cuando comienza el juego) y la bolsa debe ser capaz de contener otros objetos.

Además, el anorak y la bolsa son mucho más pesados (relativamente) que los otros objetos y hay que cambiarles su valor a 3 unidades cada uno.

Cuando se teclea A para corregir aparecen 3 valores; uno es el número del objeto, otro es el peso unitario del objeto, y finalmente el tercero son los otros atributos, que pueden ser:

0 = ninguno.

1 = contenedor para otros objetos.

2 = el jugador puede ponérselo y quitárselo.

3 = un contenedor que puede ser puesto y quitado (ejemplo: unos pantalones que tienen un bolsillo).

DECIR PAJARO SAME 12 38

MESSAGE 8

DONE

DECIR - MESSAGE 23 ;Que pregunte: ¿Quién?

DONE

No ponemos nada para la preposición A, esto permite al jugador recortar su orden si así lo quiere. En líneas generales no se debe de hacer una comprobación para una Sentencia Lógica extendida, a menos que sea necesario para diferenciar entre dos frases similares.

Como prueba final, ahora debes jugar la aventura intentando atar al perro, hablarle y todas esas chorradas. Por ejemplo, cuando estemos en el camino cerca del banco del parque, con la correa y el perro, intenta PONER LA CORREA AL PERRO Y ATARLO AL BANCO.

Luego para desatarlo digamos: DESATA PERRO.

Si estamos arriba del árbol con la bolsa, digamos PON TODO EN LA BOLSA Y SUELTA LA. BAJA Y MIRA DENTRO DE LA BOLSA.

Para hacer que el perro se siente tecléa: DECIR AL PERRO "SIENTATE".

Para hacerle volver: DECIR AL PERRO "VEN AQUI".

¿Funciona?... Malegro.

MEJORAS POR TI MISMO

Antes de que continuemos con los gráficos, aquí hay unos cuantos puntos que convendría arreglar en el juego de demostración, para que sirvan como práctica en el uso del sistema.

- 1/ EXAMINA debe responder a todos los objetos aunque sea con una respuesta general como "No veo nada especial en el _".

VOCABULARIO

Tiene un menú más complejo que los que hemos visto hasta ahora, pero la mayoría de las entradas son sólo recordatorios de las opciones a tu disposición.

El Vocabulario es una lista de las palabras que PAW puede reconocer si el jugador las teclea durante el juego. Por lo tanto, cualquier palabra que no esté en esta tabla no tendrá ningún efecto.

Hemos dejado un vocabulario inicial que ya contiene los verbos, nombres, etc., más usados en las aventuras.

Cada entrada para una palabra consiste en un máximo de cinco letras que, o bien cubren la palabra completa, por ejemplo NORTE, o son las cinco primeras letras de otra palabra más larga, por ejemplo describir, seguido del valor para esa palabra y del tipo de palabra (por ejemplo: Nombre, Verbo, etc.).

El usar solamente cinco letras para guardar una palabra reduce la cantidad de memoria requerida para guardar el vocabulario entero; también reduce la longitud de los INPUTS del jugador, y acelera la búsqueda de palabras por el propio PAW. Además, cinco letras son suficientes para establecer una diferencia entre la mayoría de las palabras usadas corrientemente.

El menú permite la INSERCIÓN y el BORRADO (delete) de palabras; el LISTADO de entradas hechas para cada tipo de palabras, y la inspección de sinónimos.

Si tecleamos S OESTE veremos que PAW muestra el sinónimo O también.

Si tecleamos P 2 tendremos una lista de todos los nombres que hay actualmente en PAW (los números que corresponden a cada tipo de palabra se muestran al lado derecho del menú).

También encontraras que PAW tiene ya en su Vocabulario todas las direcciones del compás y algunas otras útiles.

Necesitamos ahora aumentar el número de NOMBRES para poder poner las palabras que representan nuestros objetos.

Ahora una entrada relativamente simple para que se haga cargo de:
PONER LA CORREA AL PERRO:

| | | | | | |
|-------|---------|------|--------|----|---|
| PONER | CORREA | PREP | ENCIMA | | ;Para estar seguros si el jugador teclea eso |
| | NOUN2 | | PERRO | | |
| | CARRIED | | 5 | | ;Que el jugador tenga la correa |
| | SAME | | 13 | 38 | ;Que esté en la misma localidad que el perro |
| | LET | | 14 | 1 | ;Ahora el perro tiene la correa |
| | DESTROY | | 5 | | ;por lo tanto, el jugador r |
| | MESSAGE | | 21 | | la puede tener |
| | DONE | | | | ;Se le dice. |

En las entradas que siguen, introduciremos un nuevo concepto. Se trata de la modificación de la Sentencia Lógica actual.

Nosotros queremos que el juego entienda ambos, ATAR EL PERRO AL BANCO Y ATAR LA CORREA AL BANCO puesto que son la misma cosa. Pero tenemos que correa y perro tienen diferente valor como palabras, así que la entrada de ATAR PERRO irá primero en la tabla (puesto que su valor es más bajo que correa). Por lo tanto, hay que convertir el Nombre en CORREA (55) y permitir que PAW lleve a cabo una entrada que equivalga a ATAR CORREA. Un sistema similar hay que utilizar para DESATAR.

Esto es bastante importante, y lo utilizarás en tus juegos con bastante frecuencia. Hay que poner las entradas:

ATAR PERRO LET 34 55 ;La bandera 34 es el Nombre de la SL

| | | | | |
|------|--------|---------|-------|-----------------------------------|
| ATAR | CORREA | PREP | A | |
| | | NOUN2 | BANCO | |
| | | AT | 4 | |
| | | SAME | 13 | ;Donde está el banco |
| | | EQ | 14 | 38 ;El perro está aquí |
| | | PLUS | 14 | 1 ;Tiene la correa puesta |
| | | MESSAGE | 22 | 1 ;Ahora atado al banco |
| | | DONE | | ;Se informa al jugador sobre ello |

MANUAL INTRODUCTORIO - 76

| | |
|----|------------|
| 50 | ANTORCHA |
| 51 | BOLSA |
| 52 | EMPAREJADO |
| 53 | MANZANA |
| 54 | BIJETERE |
| 55 | PIEDRA |
| 56 | ANORAK |

2). Por ejemplo, tecléa [I ANTOR 50 2], etc.

Con P 2 se puede comprobar que estos nombres están ahora en el Vocabulario.

También necesitaremos usar adjetivos para diferenciar entre las dos antorchas. Con P 3 podemos ver los adjetivos que PAW va conoce.

Si no están ENCENDIDO y APAGADO insértalas procurando darles un número alto, pues son poco usados, y así PAW no tendrá que pasar a través de ellos cada vez que busque una palabra más común, por ejemplo un nombre o un verbo.

Todos los números desde el 2 al 254, están disponibles para cualquier tipo de palabra y no hay limitación en el uso de sinónimos.

Si tratas de insertar una palabra que ya esta presente, PAW te lo avisará, lo mismo que si tratas de borrar una que no esté presente.

Ten en cuenta que PAW solo tomará las primeras 5 letras cuando te refieras a una palabra e ignorará el resto.

Luego volveremos al Vocabulario, pero ahora vamos a decirle a PAW qué palabras definen nuestros objetos. Porque hasta ahora hemos descrito cada objeto, cuanto pesa, y dónde comienza, pero no le hemos dicho a PAW qué palabra identifica cada objeto.

Volver al menú principal.

La última entrada se usa para obviar el vocabulario tan limitado que entiende el perro, haciéndole que para cualquier otra frase mueva la cola, por eso se pone la última con _ _ .

Recordar que PAW va buscando en toda la tabla de Procesos, y si no encuentra una igual a la que busca caerá en esta última entrada de " _ _ .

PARSE

PARSE le permite a PAW continuar buscando conductos si no encuentra una Sentencia válida. Hay que tener cuidado aquí, puesto que la actual Sentencia Lógica puede estar un poco enrevesada (el parser intentará sacar algún significado de todos modos) así que normalmente se debe insertar algún mensaje como "Parece no entender" o algo similar, y después un DONE para volver a la acción anterior.

Si se forma una Sentencia válida PAW empezará a buscar las siguientes entradas por una similar, como hacía en la tabla de Respuestas. PARSE debe ser solamente usada en un sub-proceso que sea llamado desde la tabla de Respuestas, puesto que no tiene ningún significado en otra tabla.

Date cuenta de como las entradas VEN y AQUÍ se hacen cargo de una cantidad de frases que el jugador puede usar después de que el perro esté sentido, porque ambas tienen, una delante y otra detrás, la palabra comodín, o sea, la raya.

Y por último, la entrada " _ _ " coge cualquier sentencia que se haya metido en la cadena y para la cual el perro no tenga una respuesta específica.

Selecciona la tabla de Respuestas para insertar algunas entradas extras que controlen el habla y el dejar caer objetos del árbol.

DIAGNOSTICO

Con T del menú principal se selecciona la opción de probar el juego, o sea, hacer un test de la aventura.

Primero te pregunta si quieres o no quieres diagnóstico, si tecleas "si", aparecerá el título y la introducción, y en la parte de abajo se pide un INPUT al jugador.

El que hayas contestado que quieres diagnóstico no tiene efecto aparente todavía, pero si tecleas ENTER antes de cualquier otra cosa, el cursor desaparece y aparecerá una línea parecida a:

Flag 38 = 0 ?

con flag se refiere PAW a una bandera, en este caso la 38, que es la que lleva el número de localidad presente.

PAW contiene 256 de estas banderas, y en cada una se puede situar un número de 0 a 255. Se usan para indicar el estado en alguna parte del juego.

Por ejemplo, si decides que la bandera 11 se ponga en 1 cuando la puerta del parque esté cerrada, y haga un clear a 0 cuando esté abierta, esto servirá de marcador para que luego puedas dejar pasar o no al jugador cuando teclee ENTER.

Es decir, si la bandera estuviera a 0 podría pasar, y si la bandera estuviera a 1 le darías el informe de que la puerta está cerrada.

PAW tiene para su uso específico varias banderas (de la 0 a la 10 y de la 29 a la 59). Por eso, si el valor que aparece al lado de la bandera 38 es igual a 0, PAW sabe que tu localidad actual es 0 en este caso.

Para ver cómo funciona esto, teclea ENTER de nuevo (con ENTER cambiamos el uso entre la opción diagnóstico o continuar con el INPUT si no has tecleado ningún otro comando) y vámonos hacia el principio del juego usando la palabra NORTE o cualquier otra dirección que hayas puesto.

Similarmente a como hicimos con el pájaro, se requieren entradas en la tabla de Procesos 1, para informarle al jugador que el perro anda por ahí rondando:

```

- PERRO SAME 13 38 ;¿El perro está en la misma localidad?
  MESSAGE 17 ;Díselo al jugador.
  EQ 14 1 ;¿con la correa?
  MESSAGE 19 ;si, díselo al jugador.

- PERRO SAME 13 38
  EQ 14 2 ;¿Está el perro amarrado al banco?
  MESSAGE 20

- PERRO SAME 13 38
  GT 14 2 ;255 es mayor que 2 así que
  MESSAGE 24 ;Dile al jugador que el perro está
  sentado.

```

Ya que estamos en la tabla de Procesos 1, modifiquemos la entrada * para que también tenga [LET 13 2] (antes del GOTO), esto hace que el perro empiece en la parada de autobús.

Ahora, para que el perro asuste al pajarito, necesitamos una entrada extra en la tabla de Procesos 3. Pero debe de ir delante de la entrada que decide que el pájaro deje caer el billete y después de la entrada que hace que el pájaro vuele. Así nos aseguramos que el pájaro volará con el billete si lo tiene y lo deja si no lo tiene.

Por lo tanto, necesitamos insertar otra entrada antes de la sexta. Usaremos [I _ PAJARO 6] para hacerlo.

```

- PAJARO SAME 12 13 ;El pájaro y el perro estén en la misma
  localidad.
  LET 12 8 ;Sólo en el pabellón de música
  LET 5 3 ;Muévelo a la rama, espera 3 frases.
  AT 5 5 ;¿Está el jugador en el Pabellón de
  Música.
  MESSAGE 13 ;Dile que el pájaro se ha ido..

```

Esta última orden ha hecho que la posición del saco haya cambiado de la localidad 2 que era la parada del autobús a la localidad 254 (que es una localidad especial para los objetos que puedan ser llevados pero no puestos).

Fíjate que no hay ningún cambio en la tabla donde se localizan los objetos al comienzo, sino que lo que ha cambiado es la copia que se ha hecho de la base de datos cuando se empieza el juego.

Si ahora miras el valor de la bandera 1 (fíjate también como cuando se selecciona diagnóstico estará en uso la última bandera que hayas mirado) verás que su valor ha aumentado a 4.

Ahora prueba "quitar anorak" y recibirás el mensaje "No puedo quitarme el anorak, mis manos están llenas". Esto es porque PAW inicialmente (si tu no has dicho lo contrario) sólo permite al jugador llevar 4 objetos al mismo tiempo, y en algunos casos esta limitación puede impedirle al jugador quitarse vestido, etc., (de hecho, quitarse una prenda es cambiar la posición de ese objeto de la localidad 253, que es llevado encima, a la localidad 254 que es llevado en la mano).

Si, por ejemplo, dejamos el saco primero y tecleamos "quitar anorak" veremos que si se puede. Si miras otra vez a la bandera 1 te darás cuenta de que todavía tiene el número 4, esto es porque al quitarte el anorak has aumentado el número de objetos que llevabas en tus manos.

Ahora como prueba teclea las siguientes órdenes y fíjate qué hace cada una (usando las banderas):

```

Coger Bolsa
Quitar Anorak
Poner Anorak
Coger Manzana
Coger Billeto de autobús

```

Mensaje 18

El perro me sigue moviendo la cola.

Mensaje 19

El perro va arrastrando la correa.

Mensaje 20

El perro está atado al banco con una correa.

Mensaje 21

Confiadamente, el perro me deja ponerle la correa alrededor del cuello.

Mensaje 22

He amarrado el perro al banco.

Mensaje 23

¿A quién se lo digo?

Mensaje 24

El perro se sienta tranquilo.

Mensaje 25

He desatado el perro del banco.

No hay verdadera necesidad de hacer la rutina para el perro en una tabla de Procesos separados, porque solamente hay una entrada, pero lo haremos por si quieres ampliar el juego más tarde.

La bandera 13 contendrá la localidad actual del perro.

Es importante que árbol, rama y hoja reciban el mismo número, porque no intentamos que sean manipulables, sino que solamente las entienda el PARSER. Si fueran manipulables, tendríamos que darles números diferentes. Esto es una consideración importante de diseño.

Ahora puede ser el momento de hacer otro test del juego para ver que la orden "coger puerta" produzca una respuesta correcta.

De momento hemos creado las localidades, las hemos conectado entre si, hemos creado y descrito los objetos, les hemos asignado una palabra desde el vocabulario, un punto de comienzo en el juego, un peso relativo, si eran llevables o quitables, y si eran contenedores.

El próximo capítulo tratará de cómo se crean algunos problemas y otros personajes para hacer que el juego sea más interesante.

COGER BILLETE SAME 12 38 ;¿Está el pájaro en la misma localidad?
 ISAT 4 252 ;¿Con el billete en el pico?
 CLEAR 5 ; Esto lo fuerza a volar
 NOTDONE ; "No puedo hacer eso".

Esta entrada se disparará antes de la entrada GET _ y previene la respuesta "No hay uno de esos aquí" si el pájaro está presente con el billete.

CLEAR

Es una acción que va seguida por un número de bandera, y que pone esa bandera a 0. En este caso hará que el pájaro salga pitando, simulando su miedo a una gran mano que desciende sobre él para coger su preciada posesión.

NOTDONE

Es una acción similar a DONE, pero que engaña a PAW pensando que no has hecho nada y por lo tanto hace imprimir el mensaje "No puedo hacer eso".

Y ahora, vamos al momento de la verdad. Hay que probar el juego para ver si el pájaro vuela del pabellón de música a la rama. Juega un rato para ver si el pájaro continúa con su existencia vagabunda. Después trata de dejar caer el emparedado en la misma localidad. Date cuenta, de que si no coges el billete antes de que el pájaro se pise, el pájaro lo cogerá otra vez.

Y menos mal que acabamos con el bendito pájaro. Es complicado, pero te enseñará un montón sobre el PAW.

EL PERRO

Este perro lo ponemos para complicar el juego un poco más. Simplemente seguirá al jugador donde quiera que vaya y espantará al pajarito. Como no se espera que el perro se suba al árbol, debemos impedir que el jugador pueda tentar al pájaro con el emparedado desde la rama. Para hacer eso dispondremos que cualquier objeto que se suelte desde la rama del árbol caiga hasta el suelo.

Las dos palabras (porque _ es una palabra), indican el verbo y el nombre respectivamente. Como I es un nombre convertible en verbo (como vimos en la sección de Vocabulario) esto significa que si se teclea por sí solo, es decir, si es la única palabra que ha tecleado el jugador en la frase, será tomada como un verbo en la SL. La línea (_) indica que el nombre no es importante en esta entrada, significaría como el "no palabra" que vimos en la tabla de objetos y palabras.

Lo que esto significa es, que si el jugador teclea I sólo, PAW lo equipará con la primera entrada de la tabla de Respuestas que tenga la I, y ejecutará la orden que se describa al lado de ella.

Para ejecutar una sentencia, PAW tiene que ejecutar cada uno de los condacc (órdenes) en una lista que ya tiene. Ahora viendo en nuestro ejemplo la palabra I solamente tiene un condacc.

INVEN : es una acción (la parte activa de un condacc!). Y es la parte activa porque se encarga de hacer una lista de los objetos que el jugador esté llevando consigo o tenga puestos encima, en la pantalla. No te preocupes de momento como hacer esto, sólo nos interesa saber que lo hace.

Cuando tecleaste I (o INVENTARIO si lo has puesto como sinónimo) durante la prueba del juego, fue esta entrada en la tabla de Respuestas la que causó que pasase algo, porque se creó por el PARSE una sentencia lógica de "I _", que entonces PAW emparejó con la primera entrada en la tabla-respuesta.

INVEN, después de que ha hecho una lista de todos los objetos que lleves, ya le dice automáticamente a PAW que ha hecho algo, y cuando PAW descubre esto, busca en el PARSE otra sentencia lógica, la cual el PARSE ofrece, decodificando las siguientes frases en el INPUT del jugador. PAW entonces coge esta nueva SL y la compara con todas las entradas que hay en la tabla de Respuestas, y así sucesivamente.

- PAJARO EQ 5 3 ;¿Acaba de volar el pájaro?
 SAME 12 38 ;¿Está en la localidad del jugador?
 AT 5 ;¿o en el pabellón de música?
 MESSAGE 11 ;Ha aterrizado en la hierba.

- PAJARO EQ 5 3 ;¿Acaba de volar el pájaro?
 SAME 12 38 ;¿Está en la localidad del jugador?
 AT 8 ;¿o está en la rama?
 MESSAGE 12 ;Aterrizó en la rama.

Ahora, si el pájaro tiene el billete en su pico debemos decirselo al jugador.

- PAJARO EQ 5 3
 SAME 12 38
 ISAT 4 252 ;¿El billete no ha sido creado?
 MESSAGE 10 ;Tiene el billete en su pico.

ISAT

Es una condición seguida de un objeto y el número de una localidad y es positiva si el objeto está en la localidad especificada.

Finalmente, si el emparedado está en la misma localidad que el pájaro, el pájaro dejará el billete para coger el emparedado. Esta entrada no tiene nada que ver con la bandera 5, así que será buscada cada vez que PAW vaya al Proceso 2. Por lo tanto, si el jugador deja caer el emparedado después de que el pájaro haya llegado, la secuencia correcta aún se llevará a cabo.

- PAJARO COPYOF 2 11 ;Emparedado
 SAME 11 12 ;¿En la misma localidad que el pájaro?
 ISAT 4 252 ;¿Lleva el billete en el pico?
 COPYFO 12 4 ;Empieza a caer el billete.
 SAME 12 38 ;¿Está el jugador también por ahí?
 MESSAGE 6 ;Díselo

COPYFO

Es una acción que copia el contenido de una bandera específica a la localidad actual del objeto específico. También hay acciones COPYFF y COPYOO que son fáciles de comprender.

Este bucle (loop) se muestra en el diagrama 4 en forma de una carta de flujo que se puede seguir desde el cuadrado marcado con "comienzo". El bucle es bastante más complejo que lo que el diagrama te pueda parecer y de hecho se da una versión más completa en la guía técnica, pero de momento el diagrama 4 nos servirá.

Te aconsejamos fervientemente que vuelvas a leer los párrafos anteriores y estudies el diagrama, hasta que entiendas como PAW trabaja sobre una sentencia lógica antes de funcionar.

Vamos a considerar la segunda entrada de Respuestas:

Hacer I INVEN

como habrás deducido hábilmente, ésta entrada toma en cuenta otra forma de la cual el jugador puede pedir una lista de objetos.

Veamos ahora la entrada que pone QUIT _ (para hacer esto basta con apretar cualquier tecla excepto BREAK, ESPACIO o N):

FIN - QUIT
 - TURNS
 END

Bien, FIN es un verbo en el vocabulario, así que como la mínima frase que PAW considerará válida es un verbo, si FIN se teclea solo entonces ya el PARSER generará una SL de "FIN _".

En su búsqueda a través de la tabla de Respuestas, PAW encontrará la entrada que hemos visto más arriba y empezará a ejecutar los condacc.

QUIT es una condición (la parte condicionante de la palabra condacc). Una condición solo decide si PAW continúa para llevar a cabo el siguiente contacto de la lista, es decir, no toma de por sí ninguna acción, sino una decisión.

QUIT lo que hace es determinar si el siguiente contacto puede ser ejecutado, preguntándole al jugador "¿estás seguro?". Si él contesta NO, entonces QUIT avisa a PAW que ha hecho "algo" y éste irá en busca de otra SL (dejando en paz QUIT).

Primero hay que determinar si el pájaro va a salir pitando, esto pasará cuando la bandera 5 tenga valor 0 (empezará a contar desde 3). Entonces, si el billete está en la misma localidad que la del pájaro, será destruido (puesto que la localidad 252 indica que el pájaro lo tiene) y si el jugador está en la misma localidad que el pájaro, se le dirá que el pájaro se ha llevado el billete.

Tienes que darte cuenta de que el pájaro sigue con sus ciclos de movimientos aunque el jugador no lo vea. De hecho en el mundo de PAW un árbol se cae aunque no haya nadie para verlo. Así de real es.

```
- PAJARO COPYOF 4 11 ;Copia la localidad del objeto 4 (el
  billete) a la bandera 11.
  SAME 11 12 ;Y mira si está en la misma localidad
  que el pájaro.
  ZERO 5 ;¿Va a volar el pájaro?
  DESTROY 4 ;El pájaro coge el billete.
  SAME 12 38 ;¿Está el pájaro en la misma localidad
  que el jugador?
  MESSAGE 7 ;Díselo al jugador.
```

Fíjate, no hay acción DONE porque queremos que PAW haga cada entrada, una detrás de otra, siempre que se cumplan los requisitos de las condiciones de ellas. La tabla anterior te muestra como se pueden mezclar condiciones y acciones para crear nuevas condiciones.

COPYOF

Es una acción que debe ser seguida por el número de un objeto y una bandera. Lo que hace es que copia la localidad actual del objeto que se especifique, en la bandera especificada. La usamos en esta situación para ver si el billete está en la misma localidad que el pájaro.

SAME

Es una condición que compara el contenido de dos banderas, y que es favorable o positiva si ambas son del mismo valor.

Pues bien mi querido saltamontes, la razón es que, aparte del hecho de que es mucho más fácil ponerlas en una tabla que en propio juego, **TU JUEGO A LO MEJOR NO LAS NECESITA**, y de esta forma las puedes borrar cuando quieras.

DESC es una acción. De hecho, es la que se usa en la entrada "M" de la tabla. Y esta acción hace que PAW abandone la búsqueda por la tabla de respuestas y reDESCriba la localidad actual del jugador. También tiene varios sinónimos: mirar, describir, etc., en el Vocabulario.

SAVE Y LOAD son dos acciones que permiten hacer un SAVE y un RELOAD del estado actual del juego en cinta. La posición actual del juego incluye también cualquier otra pieza de información que se necesite para restaurar exactamente el juego en el mismo estado en que estaba, ello incluye los valores de las banderas, las posiciones de los objetos, y mucha más información.

No debes confundir aquí los anglicismos SAVE y LOAD que hemos dejado en el vocabulario, con las acciones SAVE y LOAD. Nosotros hemos incluido también otros sinónimos, como GRABAR Y CARGAR, en el Vocabulario, pero todos ellos usarán también las acciones SAVE Y LOAD en la tabla de Respuestas.

Es de notar que ambos SAVE y LOAD ya de por sí hacen una acción DESC cuando han terminado. Lo cual significa que cualquier condacc que siga, será ignorado igualmente en el INPUT del jugador.

RAMSAVE Y RAMLOAD son dos acciones similares a SAVE Y LOAD, excepto que ellos usan un "buffer" (área de memoria libre) para guardar la posición del juego. Esto significa que no tienes que andar toqueteando cintas.

Solamente una posición puede ser guardada, y como la guarda en la memoria, suponemos que sabes que si apagas el ordenador desaparecerá. Aunque de todos modos, esto debes hacérselo ver claramente al jugador en las instrucciones.

También el área del buffer se pierde cuando vuelves al editor, porque tú puedes desear cambiar algún diseño en el juego entre dos textos.

Mensaje 13

El pajarito ve al perro y sale pitando.

Mensaje 14

El pajarito se va.

Luego pondremos los mensajes que vaya a usar el perro, pero ahora debemos seleccionar la tabla de Procesos y prepararnos para otras de las virguerías de PAW.

SUB-PROCESOS

En una aventura de verdad que contenga varios PSI y un montón de acciones, las tablas de Procesos 1 y 2 pronto acabarán llenas, y se hace cada vez más difícil trabajar de esta forma. Entonces viene el momento de dejar que las otras tablas de Procesos comiencen a actuar. Estas pueden ser llamadas desde las tablas de Procesos 1 y 2 o de la tabla de Respuestas, y usadas como una extensión de la tabla desde la cual fueron llamadas.

El llamar a otro proceso hace que PAW guarde en memoria dónde estaba en el momento de ser llamado, y continuará actuando en la misma forma en que lo estaba haciendo. Por ejemplo, si se llamó desde una tabla de Respuestas, PAW tratará de hacer coincidir la Sentencia Lógica actual con cualquier otra entrada. Pero si es llamado desde una tabla de Procesos, sea la 1 ó la 2, PAW sencillamente ejecutará cada entrada. (ESTO ES MUY IMPORTANTE).

Por otra parte, cuando algo es DONE en el Proceso llamado, PAW volverá a la tabla original y continuará trabajando. Por lo tanto se pueden hacer acciones bastantes complejas mediante el uso de los sub-procesos. Los usuarios que sepan algo de programación reconocerán fácilmente que son sencillamente sub-rutinas.

Cuando PAW está en un sub-proceso, es posible que sea llamado para actuar en otro sub-proceso (¿un sub-sub-proceso?), y así se puede continuar con sub-sub-sub hasta un valor de 10.

NO PRESENTES: ¡En cualquier otro sitio!. Esto también incluirá la localidad 252, que es una localidad imaginaria donde todos los objetos que aún no existen son guardados.

Resumiendo:

Aquí = Localidad 255
Llevados = Localidad 254
Puestos encima = Localidad 253
No presentes = Localidad 252 = No creado todavía.

Vamos a ver otras dos entradas en la tabla de Respuestas:

COGE TODO DONALL 255 (Parámetros: Localidad donde estás: Aquí)

COGER - AUTOG
DONE

éstas dos entradas son las que permiten al jugador coger cualquier objeto. Coger un objeto significa cambiar su localidad desde AQUÍ (255) a LLEVADO (254).

Ignoremos por ahora la entrada "coge todo" y vamos a estudiar la entrada "COGER".

Como dijimos antes, esa línea baja significa "cualquier palabra", así que no importa cual nombre teclee el jugador en una frase que contenga el COGER. La entrada COGER _ siempre casará (esto es lo que se llama "cargando" o "preparando" una entrada).

Veamos la frase "coger la manzana": la será ignorada por PAW porque no está en el Vocabulario, pero la SL será "coger manzana", y esto "cargará" la entrada "GET _" cuando PAW encuentre este conducto.

AUTOG es una acción que AUTOMáticamente cogerá cualquier objeto que haya sido especificado por el nombre.

Es donde la tabla de Palabras-Relacionadas con Objetos viene a funcionar. AUTOG mira a través de esa tabla buscando una entrada que sea similar al nombre de la SL, cuando la encuentre (manzana en nuestro ejemplo) entonces sabe el número del objeto al cual se refiere (que en el caso de la manzana es 3).

El juego de práctica que hemos hecho es bastante fácil de resolver (bastante tontito, la verdad). Así que le vamos a añadir un poco de complejidad creando dos caracteres que vayan vagando por nuestro pequeño mundo. Estos caracteres son llamados PSI, o Caracteres Seudo Inteligentes, porque es obvio que aunque no pueden pensar, deben de parecerle al jugador como si lo hiciesen.

Un PSI consiste principalmente en una colección de mensajes, banderas y entradas en tablas de procesos. Verás que con unas pocas entradas simples se pueden crear efectos sorprendentemente real.

La creación de un complejo PSI puede tomar mucho más tiempo, pero en general sigue los mismos principios que los que vamos a hacer con los nuestros: un pajarito y un perrito.

El pájaro se pone para complicar el escenario de la siguiente manera: Cogará el billete al principio del juego (normalmente se le daría una localidad no usada como contenedor de los objetos que lleve el pájaro, pero nosotros usaremos la localidad 252, puesto que tenemos solamente un PSI que puede tener un objeto).

De modo que el jugador debe convencer al pajarito para que deje caer el billete. El intentar quitárselo o cogerlo dará el mensaje "No puedo hacer eso", y que el pajarito se vaya volando hacia otra parte.

El pajarito también dará la paliza volando entre el pabellón de música y una ramita del árbol, y esto lo hará a intervalos regulares. La única manera de convencer al pajarito para que deje caer el billete de autobús es dejar caer el emparedado en la misma localidad.

Lo primero que hay que hacer es ir a la tabla de Objetos Inicialmente En [la tabla I] y cambiar el objeto 4 a no creado, es decir, darle al objeto 4, que es el billete, el valor de 252.

Esto hace que el objeto no exista como un objeto, porque lo lleva el pajarito en el pico.

Explicaremos el mecanismo:

Si el jugador teclea la frase COGE TODO, el PARSER creará una sentencia lógica (SL) de "COGE TODO" y entonces hará pareja con la entrada presente en esta tabla y PAW pasará a ejecutar la acción DOALL.

DOALL es una acción que debe ser seguida con un parámetro. El parámetro nos da el número de la localidad que usaremos.

Lo que hace DOALL es buscar la lista de localidades en la cuál se encuentra cada objeto, intentando encontrar entradas que sean iguales para el parámetro que se ha dado (que en este caso es 255, una localidad especial que significa que se use la localidad en la cual el jugador esté presente).

Cuando encuentra uno similar, o el mismo, entonces se va a la tabla objetos relacionados con palabras para encontrar la palabra del Vocabulario que describe el número del objeto.

Esta palabra se pone en la actual SL (reemplazando el nombre todo) y una bandera se resetea para indicar que DOALL está activo.

Luego PAW sigue mirando el resto de la tabla de Respuestas buscando una entrada que haga juego con la SL recientemente modificada.

Esta entrada será COGE (de la cuál hemos hablado antes), que cogerá el objeto que PAW haya buscado en la tabla de Objetos.

Cuando haya terminado de hacer lo anterior, PAW se dará cuenta de que todavía permanece activo el DOALL, o sea, la bandera está seteada y vuelve otra vez a la entrada COGE TODO.

En realidad lo que hace es saltar directamente a la acción DOALL y buscar otro objeto para generar una nueva SL, y así sucesivamente para todos los objetos que estén en la localidad especificada.

Cuando se acaban los objetos, entonces la bandera se resetea para demostrarle a PAW que no está activo el bucle y se le dice a PAW que busque otra SL.

Combinado con la entrada previa tiene el efecto de hacer que PAW pare de buscar en las tablas de contacto a menos que la Sentencia Lógica fuera PONER _ EN o DENTRO DE LA BOLSA donde la raya baja sería cualquier objeto.

AUTOP

Debe ser seguida por el número de una localidad. Nosotros hemos puesto la localidad 1 aparte con un propósito especial desde el comienzo de este ejemplo. Esto es, la hemos usado como si estuviera dentro de la bolsa. Por lo tanto AUTOP, de la misma forma que AUTOD, busca en la tabla de Objetos-Palabra por un nombre que haga pareja con el primer nombre de la Sentencia Lógica. Cuando ha encontrado uno, lo pone en la localidad que le hemos dado, respondiendo "He puesto el o la _ en o dentro de la bolsa".

La entrada DELTA TODO ya existente, también se hace cargo de PONERLO TODO EN LA BOLSA, porque no especifica que EN LA BOLSA sea parte de la Sentencia Lógica, por lo tanto se disparará en ambas ocasiones, y en ambos casos 254 es la localidad desde la cual vendrán los objetos.

Ahora para COGER un objeto FUERA DE LA BOLSA (o sacar un objeto de la bolsa), debemos de teclear una entrada similar, que haga nula la entrada ya presente de COGER _ . Así que tecleemos [I COGER _ 0 ENTER], con ello nos aseguramos que será puesta antes.

PREP FUERA NOUN2 BOLSA PRESENT 1 AUTOT 1 DONE

AUTOT

Debe ser seguido por un número de localidad, que debe ser de la cual venga el objeto que se va a sacar. (En éste caso la 1).

Por otra parte, para hacer la versión SACAR TODO necesitamos también otra entrada. De momento COGER TODO causa un DOALL 255, lo cual es la posición actual del jugador. Para que saquemos todo de la bolsa necesitamos generar un contador de los objetos que estén dentro de ella (localidad 1), así que hay que insertar una entrada nueva de coger todo, que salte por encima de la existente.

El que un objeto sea examinable, solamente requiere que el escritor haya dejado un mensaje que dé más información sobre ése objeto, por ejemplo, en el caso de la manzana diríamos "La manzana está madurita y muy sabrosa".

Cambia el texto del mensaje 0 (A 0 ENTER) e inserta los siguientes mensajes para poder examinar todas las otras cosas del juego.

Mensaje 1

Es un emparedado delicioso de jamón y queso.

Mensaje 2

El billete tiene un número y el nombre de la compañía de transportes para aventureros impreso.

Mensaje 3

El banco está anclado firmemente a una base de concreto.

En nuestro juego de práctica solamente usamos cuatro objetos, pero en los juegos más complicados hay que poner muchos detalles para muchas cosas, aunque no sean muy importantes en el juego, porque ésto le da un toque de realismo que hace que el jugador se sienta atraído o envuelto por la atmósfera.

Así que volvamos a la tabla de RESPUESTAS (R) y a trabajar.

Vamos a comenzar con la manzana. La frase que el jugador tecleará será "EXAMINAR LA MANZANA" (o EXAMINA MANZANA si es bastante vago), y esto producirá una SENTENCIA LÓGICA de "EXAMINAR MANZANA". Así que necesitaremos poner una entrada con estas dos palabras.

Tecléa (I EXAMINAR MANZANA ENTER), PAW ignorará las letras que sobren (que sean más de cinco), e imprimirá la entrada en la parte de arriba de una pantalla limpia y esperará a que tu teclées la lista de contactos para esa entrada.

ANYKEY

Imprime "Pulse tecla para continuar" en la parte inferior de la pantalla, y espera a que pulses una tecla antes de permitirle a PAW continuar.

GOTO

Debe ser seguido por un número de localidad y mueve al jugador a esa localidad (efectivamente, pone la bandera 38, la bandera que marca la localidad actual del jugador, al valor dado). Como no hace nada más, debe ir seguida por un DESC para que PAW describa la nueva localidad.

Esta entrada entonces lo que hace es que la pantalla del título aparezca, luego espera por el toque de tecla para pasar al juego en la localidad correcta.

Ahora es el momento de ir a la tabla de conexiones y quitar la entrada que pusimos desde la localidad 0 porque ya no se necesita.

Hagamos un test del juego para ver las dos entradas en acción. Si teclésemos de una sola vez toda la frase:

OESTE, OESTE, ARRIBA. COGER EL BILLETE. ABAJO, ESTE Y ESTE.

tendríamos el juego terminado, con su mensaje de si quieres volver a jugar otra vez. Si no es así, vuelve a chequear las entradas en las tablas de Procesos 1 y 2.

OBJETOS DENTRO DE ALGO

Vamos a ver la habilidad de la bolsa para contener objetos. Ello requiere algunas entradas en la tabla de Respuestas, porque vamos a permitir al jugador que MIRE DENTRO DE LA BOLSA, así que necesitamos un nuevo mensaje: "En la bolsa hay:". Por lo tanto, selecciona la opción de MENSAJES e INSERTALO (debe ser Mensaje 5).

Otra vez a la tabla de RESPUESTAS.

Vamos a dotar al jugador de la opción de decir "Poner todo dentro de la bolsa", además de "poner (objeto) en la bolsa".

Esto forzará a PAW a dibujar cualquier gráfico que exista en esa localidad y al mismo tiempo hacer una descripción. Sobre esto entraremos con más detalle cuando hablemos de los gráficos.

Si asumimos que la manzana está realmente presente, entonces PAW continuará con el siguiente condacc y desplegará nuestra descripción de la manzana (MENSAJE 0), luego, la acción DONE le dice a PAW que vaya y busque otra Sentencia Lógica, porque ya ha hecho algo (esto previene que continúe y pase lo que pasó antes, es decir que coja la siguiente entrada que es LOOK).

Si hay una entrada que es incorrecta, se puede corregir tecleando por ejemplo, A MIRAR MANZANA. Sin embargo, en las tablas de Procesos puede haber más de una entrada que tenga los mismos valores de palabras, y entonces todas ellas se verán presentadas una a una para que se pueda corregir.

Si no quieres corregir una en particular, sencillamente teclea ENTER y déjala tal como está.

Para borrar una entrada completamente de una tabla, lo que hay que hacer es quitar todos sus condaccs.

Por ejemplo, corrige la entrada con A, pulsa la tecla EDIT dos veces para limpiar el buffer y pulsa ENTER. Con eso la entrada quedará completamente borrada.

Pero de momento, si el jugador trata de EXAMINAR cualquier otra cosa excepto la manzana (o cuando trata de EXAMINAR LA MANZANA y ella no está presente) entonces, como vimos antes, se le recompensará con una descripción de su localidad actual.

Vamos a insertar las entradas para poder examinar el emparedado, el billete de autobús y el banco.

Las entradas las listaremos como las verías si usaras la opción P después de que fueran tecleadas, y pondremos al lado algunos comentarios solamente para referencia. Pero no te olvides de que hay que meterlas como hiciste para introducir EXAMINAR MANZANA anteriormente.

WEST TIME

Imprime espacios hasta el final de la línea actual. Esto permite que el mismo color de Papel continúe hasta el final de la línea sin tener que teclear todos los espacios. Su principal objetivo aquí es asegurar de que cualquier texto que se imprima lo sea en la siguiente línea, porque PAV no pone automáticamente una línea en blanco después de la descripción de las localidades. La guía técnica nos mostrará como usar este efecto para modificar una descripción de localidad, para reflejar los cambios que haya.

De ahora en adelante las dos entradas deben de ser consideradas como una pareja, su propósito final es hacer una lista de los objetos que estén en la localidad actual. Veamos como:

PAW usa la bandera \emptyset para determinar si hay luz para que el jugador pueda ver (esto no ha sido utilizado de momento en nuestra demostración), si no, hay luz, la bandera tendrá un valor diferente de \emptyset y PAW contestará "está muy oscuro para ver" en vez de hacer una descripción de la localidad. En este caso los objetos que están presentes no deben de aparecer en lista puesto que se supone que el jugador no los puede ver.

El objeto \emptyset siempre es tomado por PAW como el objeto que provee luz (objeto \emptyset = fuente de luz), esto es por lo que en nuestro juego de demostración, el objeto \emptyset es una antorcha encendida. Si este objeto está presente cuando en el juego se supone que está oscuro (o sea, que la bandera \emptyset no sea \emptyset) entonces, la sola presencia del objeto \emptyset tendrá preferencia sobre el hecho de que no haya luz, y los objetos deben ser descritos.

Estos dos últimos párrafos son muy importantes y debes releerlos hasta estar seguro de que los entiendes.

Así pues, vemos que las dos entradas nos dan un ejemplo de cómo se usa PAW para crear una situación OR. Por ejemplo, haz una lista de los objetos si hay luz o (OR) el objeto \emptyset está presente.

De momento hay dos tablas de Proceso en la base de datos y PAN, tal como hacía en el de Respuestas, buscará a través de ellas, pero las buscará no después de haber obtenido una sentencia lógica, sino de la siguiente forma:

Proceso 1: El proceso 1 es chequeado inmediatamente después de que PAW ha descrito una localidad. Esto permite que la información se imprima solamente cuando el jugador llegue a ésa localidad o cuando haya obtenido una redescipción.

Proceso 2: El proceso 2 es revisado justo antes de pedir una nueva sentencia lógica al PARSEK. Es usado como si fuera el "turno" de PAW para jugar.

Estudia detenidamente las formas en que PAW busca en la tabla de Proceso 1 v en la tabla de Proceso 2, ES MUY IMPORTANTE.

La diferencia principal es que en estas tablas, PAV no intenta buscar una entrada similar a una sentencia lógica, sino que EJECUTA CADA ENTRADA.

De momento, mientras jugábamos en nuestro juego de demostración, hemos tenido que terminar siempre tecleando F11N. Ahora en la "Historia Original" lo que tenemos que hacer era ayudar al pasajero a encontrar el billete de autobús antes de que éste llegase. Es obvio que podemos poner una entrada en la tabla de Respuestas para que si el jugador teclea "coger billete" (y este estuviese presente), se termine el juego.

COGER BILLETE PRESENT 4 ; El billete está ahí
TURNS
END

pero pensamos que sería mucho más conveniente terminar el juego cuando el jugador vuelva a la parada del autobús.

Para hacer eso, primero necesitamos un mensaje que describa la llegada del autobús. Así que vamos al menú principal, seleccionemos la opción mensaje y pongamos el siguiente mensaje: