

USO DE LA LIBRERÍA "RASTROS.H"

1. Introducción

La librería "Rastros.h" ha surgido en el marco del desarrollo de las librerías de arquetipos para PNJs (Personajes No Jugadores). Para realizar las librerías de arquetipos de PNJs tuve que desarrollar una librería genérica que permitiese el tratamiento de árboles de autómatas de una forma que fuese 'heredable' desde un arquetipo a la implementación final del PNJ del juego. Así surgió la librería "Automata.h" que espero poder sacar a la luz en breve. Una de las características incorporadas a esta librería era que con cada cambio de estado el autómata en cuestión (o sea el PNJ) puede cambiar de modo de movimiento. Por ejemplo si un PNJ entra en un estado de 'pánico' al presenciar a un espantoso monstruo surgido de las entrañas de la tierra, lo normal es que salga corriendo de allí lo más lejos posible, al menos durante un tiempo. Pero los tipos de movimiento incluidos originalmente en "PNJMovil.h", eran demasiado pocos para obtener un comportamiento mínimamente interesante de los PNJs. Así nació "Moviles.h", que modifica algunas cosas de la "PNJMovil.h" original y que agrega algunas clases de movimiento adicionales. Uno de los movimientos que parecía más razonable incluir era el 'seguimiento de rastros' tanto visuales como odoríferos, de esta forma se podía poner un rastreador que siguiese las huellas de un caballo o un perro que siguiese un rastro de olor invisible para el resto de los PNJs y para el jugador. De esta enrevesada forma surgió "Rastros.h" y esperemos que el resto de las librerías mencionadas puedan ver en breve la luz.

¿Y para que sirve esta librería? Pues para crear rastros de diversos tipos, rastros visuales, rastros de olor, rastros que se disipan a varias velocidades posibles y rastros que no se disipan nunca.

2. ¿Cómo usar la librería?

Para que funcione esta librería es necesario incorporar a tu aventura una línea de include. Debe llamarse a la librería DESPUES de la librería de Gramática pero antes de las librerías que muevan tus PNJs (PNJMovil, Moviles, Automata o cualquier otra librería similar), ya que si estas librerías han de tener en cuenta los rastros deberían ver las clases de los mismos definidas antes de su código. Un ejemplo sería:

```
! Inclusión de la definición de verbos
include "Gramatica";
! Incluimos la funcionalidad de Rastros
include "Rastros";
! Incluimos la funcionalidad de Automatas
include "Automata";
```

Pero además deberás iniciar el motor de rastros para eso incluye en tu 'Inicializar' la línea:

```
IniciarRastros();
```

3. Percepción de los rastros

La librería define un conjunto de constantes que determinan los mensajes que generará ante la presencia de un rastro y si el rastro se puede percibir o no. Estas constantes son:

- **nivelPerceptible:** por debajo de este nivel el rastro simplemente no será perceptible ni por el jugador ni por los PNJs que usen los métodos de la librería aunque podría ser perceptible por PNJs que usen métodos diferentes propios (PNJs con mayor sensibilidad). El valor numérico correspondiente a este nivel de 'intensidad' del rastro es de 2.
- **nivelCasiInvisible:** por debajo de este nivel la librería indicará si no se redefine el método 'describeIntensidad' que este rastro es 'casi imperceptible'. El valor numérico correspondiente a este nivel de 'intensidad' del rastro es de 5.
- **nivelViejo:** por debajo de este nivel la librería indicará si no se redefine el método 'describeIntensidad' que este rastro es 'casi imperceptible'. El valor numérico correspondiente a este nivel de 'intensidad' del rastro es de 8.
- **nivelReciente:** por debajo de este nivel la librería no indicará nada si no se redefine el método 'describeIntensidad'. El valor numérico correspondiente a este nivel de 'intensidad' del rastro es de 11.
- **nivelClaro:** por debajo de este nivel la librería indicará si no se redefine el método 'describeIntensidad' que este rastro es 'claro'. Por encima de este nivel la librería indicará que el rastro es 'muy claro'. El valor numérico correspondiente a este nivel de 'intensidad' del rastro es de 14.

Los valores de estas constantes se pueden redefinir definiéndolas ANTES de la línea que incluye la librería, por ejemplo:

```
! Inclusión de la definición de verbos
include "Gramatica";

! Redefinición de casi todos los valores
Constant nivelPerceptible = 4;
Constant nivelCasiInvisible = 7;
Constant nivelViejo = 10;
Constant nivelReciente = 12;

! Incluimos la funcionalidad de Rastros
include "Rastros";
```

Los valores son constantes, es decir, se puede cambiar su distribución en todo un juego PERO no se pueden cambiar durante la ejecución del juego. Si no fuesen constantes no podrían usarse en las propiedades 'rastrosDejados' que se verán más adelante.

4. Tipos de Rastros

Para crear un rastro deberás determinar ante todo las características de dicho rastro. Para hacer esto debes crear una instancia de la clase TipoRastro. Por ejemplo aquí podemos ver algunos ejemplos extraídos de mi laboratorio de pruebas:

```
!
! Rastros misteriosos
!
TipoRastro RastroDeSangre "rastro de sangre"
    with
        adjetivos 'sangre',
        nombre_plural "rastros de sangre",
        descripcion "Se trata de sangre putrefacta que parece
desaparecer rápidamente. ¡Desde luego no se trata de
sangre normal!",
        multDisipacion 1;

TipoRastro RastroDePelos "rastro de pelos"
    with
        adjetivos 'pelos',
        nombre_plural "rastros de pelos",
        descripcion "Montoncitos de pelos dispersos. ¿Algo o
alguien se está quedando calvo?",
        multDisipacion 0;

TipoRastro RastroDeOlor "extraño olor"
    with
        esOloroso true,
        adjetivos 'extrano' 'azufre',
        nombre_plural "extraños olores",
        descripcion "Un extraño olor, como a azufre.",
        multDisipacion 2;
```

El primero de los rastros es un rastro visual de sangre que se disipa al ritmo básico de la librería (un punto por turno), el segundo es un rastro visual constituido por pelos y que no se disipa nunca (multDisipacion 0) y el ultimo es un rastro odorífero que se disipa al doble de velocidad que los otros.

Las propiedades incluídas en los rastros son:

- **multDisipacion:** se trata de un multiplicador que acelera la desaparición del rastro. Cosas muy 'volubles' o sutiles como un perfume deberían desaparecer muy rápido aunque fuesen muy intensos en un principio. Para conseguir este efecto pon un número alto en esta propiedad.
- **nombre_plural:** texto descriptivo (nombre corto) en el caso de que no haya un rastro de este tipo, sino varios presentes.
- **esOloroso:** indica si el rastro es de naturaleza olfativa, si es así no aparecerá en la lista de objetos visibles, pero se podrá 'Oler'.

Los métodos que pueden mencionarse por si se desea cambiar el comportamiento por defecto de un tipo de rastro son:

- **describeIntensidad:** recibe un sólo parámetro 'intensidad' e imprime una descripción correspondiente a la intensidad precedido por un espacio en el caso de que imprima algo.
- **describeRastro:** recibe los parámetros de 'origen' (lo que ha creado el rastro), 'dirección' (hacia dónde se encamina el rastro) e 'intensidad' y genera una descripción del rastro.
- **antes:** la tradicional 'antes' de todos los objetos. En este caso se han implementado respuestas para:
 - **Coger:** se dice que no se puede
 - **Oler:** se realiza el tratamiento correcto según sea un rastro de olor o no
 - **Examinar:** produce los mensajes necesarios según la presencia o no de rastros y de sus características.

Todos estas acciones reciben además dos variables genéricas que se calculan a priori:

- **padre:** la localización donde se encuentra el rastro.
- **numero:** el número de rastros de este tipo que hay presentes.

Finalmente en esta librería he tenido especial cuidado en la redefinibilidad de todos los elementos, de forma que antes de incluir esta librería se puede incluir una definición diferente de 'TipoRastro' y no se usará la definición de la librería, sin que dejen de usarse el resto de las clases incluidas.

5. ¿Dónde pueden estar los rastros?

La librería guarda los rastros en propiedades de las localidades así que no puedes dejar un rastro en cualquier lugar sin tan sólo en un 'LugarConRastro'. Para hacer que un conjunto de localidades sea un lugar por donde puede haber rastros define una clase que incluya 'LugarConRastro' y haz que todas esas localidades sean de esta clase. Un ejemplo sacado de 'gauto.inf':

```
Class Lugar
  class LugarConRastro,
  with   describir [; print (string) self.descripcion;
            print "^"; <<Salidas>>;],
  antes[;
    Examinar:
      if (uno==obj_arriba) !arriba
        "El techo está muy alto y es de piedras.";
      else if (uno==obj_abajo) !abajo
        "El suelo es de piedras y está muy frío."],
  cantidad,
  has     luz;
```

Las propiedades incluidas en los lugares con rastro son:

- **multGeneral:** permite hacer que en una determinada localización la disipación se multiplique o cese por completo (si se usa el valor cero).

Los métodos útiles de esta clase son:

- **marcaRastro:** graba un rastro en la localidad. Recibe los parámetros 'origen' (lo que deja el rastro que DEBE SER un ObjetoRastreable), 'tipo' (el objeto de la clase TipoRastro que contiene la descripción del rastro), 'direccion' (uno de los objetos de la brújula -obj_n, obj_s, etc...- indicando hacia dónde se dirige el rastro) e 'intensidad' (nivel al que se marca el rastro).
- **disipaRastros:** esta función será llamada por el demonio de la librería, pero puedes usarla para disipar rastros de forma diferente. Recibe un único parámetro, 'factor' que es un factor multiplicativo de lo que se debe quitar a los rastros. El demonio de la librería siempre llamará a esta función con 'factor' con valor 1. Si la llamas con un valor muy grande, como 100, borrarás todos los rastros excepto aquellos que no se disipan.

6. Objetos rastreables

La librería usa el concepto rastreable para aquellos objetos que normalmente generarían un conjunto de rastros y por lo tanto podrían ser hallados siguiendo alguno de los rastros que dejan. Para crear un objeto rastreable incluye esta clase en tu objeto, por ejemplo:

```
Automata Gato "Malaquías, el gato" Caja
    class ObjetoHablaante ObjetoOyente ObjetoRastreable,
    with nombre 'gato' 'Malaquías' 'Malaquías'
        descripcion "Es el gato negro Malaquías.",
        rastrosDejados RastroDePelos nivelReciente true,
        voz Maullido;
```

Las propiedades de los objetos rastreables son:

- **rastrosDejados:** es un array con tríos de valores siempre constituidos por:
 - El tipo de rastro a dejar (un objeto de la clase 'TipoRastro'.
 - Una intensidad por defecto para dejar este rastro
 - Booleano que indica si el jugador puede reconocer o no este rastro

Los métodos de los objetos rastreables son:

- **rastroReconocible:** recibe como parámetro un TipoRastro y si este tipo es de los que puede 'dejar' el objeto rastreable indica si el jugador lo reconocería o no.
- **cambiaRastro:** para el tipo indicado en el primer parámetro cambia si es reconocible o no (primer parámetro) y la intensidad por defecto (segundo parámetro que es opcional, si no se pasa no se cambiará la intensidad por defecto).

- **dejaRastros:** esta función puede tomar hasta tres parámetros diferentes: 'dirección' (obligatorio, un objeto de la brújula que indica hacia dónde se marca el rastro), 'tipo' (opcional, si no se indica el objeto rastreable marcará todos los rastros indicados en su rastrosDejados con sus intensidades por defecto) e 'intensidad' (opcional, si no se indica se usa la intensidad por defecto que corresponda).