

Máquinas de estado finitos

Una Máquinas de estado finitos (FSM, por sus siglas en inglés) es un modelo matemático compuesto por:

- Estados: Representan las distintas “modos” en los que puede estar el NPC (por ejemplo: patrullando, persiguiendo, atacando, huyendo).
- Transiciones: Reglas que definen cuándo y cómo el NPC cambia de un estado a otro (por ejemplo: si ve al jugador pasa de patrullando a persiguiendo).
- Eventos o condiciones: Son los disparadores de las transiciones (como la distancia al jugador, recibir daño, perder de vista al objetivo, etc.).
- Acciones: Lo que el NPC hace mientras está en un estado determinado (animaciones, movimientos, sonidos, etc.).

Las FSM pueden implementarse utilizando simples estructuras condicionales if-then-else (o switch/case) es lo que se conoce como “FSM implícita”. Este tipo de FSM funcionan para casos simples, pero no escala bien y puede volverse difícil de mantener, depurar o extender.

Existen otras formas más robustas, modulares y limpias de implementar las FSM, uno de los paradigmas más usados es el orientado a objetos donde cada estado es una clase, y las transiciones se manejan dentro de esas clases.

Las ventajas de este modelo es que cada estado encapsula su lógica lo que lo hace fácil de extender ya que añadir un nuevo estado solo requiere una nueva clase. Evita grandes bloques de if/else que hacen el código ilegible.

Tracedump:

```
newBaseSize: 12pt  
newBaseSizeInPt: 12
```