

1. Diagramas de estados

1.1 Introducción aos diagramas de estados

A continuación, aprenderanse os seguintes conceptos e manexo de destrezas:

- Comprender o propósito e función dos diagramas de estados.
- Manexar diagramas de estado sinxelos, interpretando correctamente diagramas xa feitos e desenvolvendo diagramas novos a partir de escenarios propostos.

1.2 Propósito e función dos diagramas de estados

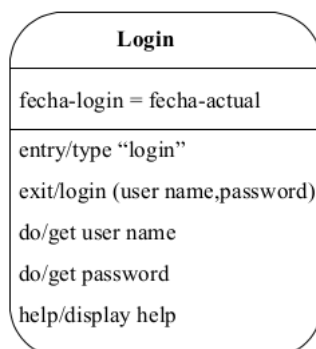
Os diagramas de estados, tamén chamados diagramas de máquina de estados, vannos servir para mostrar os diferentes estados polos cales pasa un obxecto durante o seu ciclo de vida e a maneira en que o obxecto cambia dun estado a outro en resposta aos eventos que chegan a el.

Dentro dun proxecto software, os diagramas de estado normalmente empréganse sempre para mostrar o comportamento dun único obxecto ao longo do seu ciclo de vida pero tamén poden ser útiles para representar a secuencia de eventos que poden afectar a un sistema nun determinado caso de uso. Neste caso falamos de **diagramas de estado para casos de uso** e unicamente soen empregarse en casos de uso complexos con moitos eventos que deben de sucederse seguindo unha orde establecida (por exemplo cando estamos utilizando un procesador de texto).

Elementos

Nun diagrama de estados basicamente aparecen dous tipos de elementos:

- **Estados.** Un estado é unha situación na vida dun obxecto. Identifica un período de tempo do obxecto no cal o obxecto está esperando algunha operación, ten certo estado característico ou pode recibir certo tipo de estímulos. Represéntase mediante un rectángulo cos bordos redondeados, que pode ter tres compartimentos: un para o nome, outro para o valor característico dos atributos do obxecto nese estado e outro para as accións que se realizan ao entrar, saír ou estar nun estado (**entry**, **exit** ou **do**, respectivamente).



En moitos diagramas de estado omítense os dous compartimentos inferiores.

Márcanse tamén os estados iniciais e finais mediante os símbolos ● e ⊙, respectivamente.

- **Transicións.** Unha transición é unha relación entre dous estados que indica que un obxecto que estea no primeiro estado realizará certas accións e entrará no segundo estado cando ocorra un evento determinado e se satisfagan unhas condicións específicas. Graficamente represéntase cunha liña continua con punta de frecha no estado destino e normalmente acompáñase cun texto indicando o nome do evento que desencadea a transición.

Pode existir unha transición que teña o mesmo estado orixe e destino.

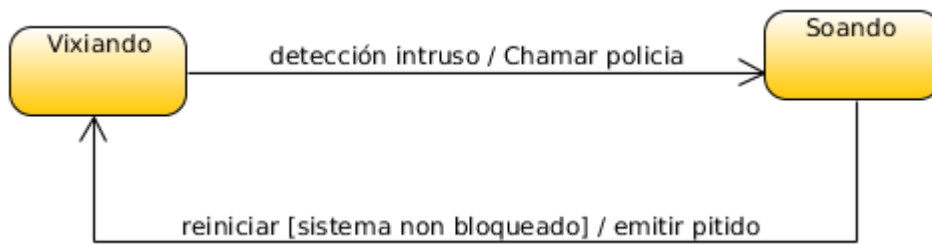
O nome do evento pode vir acompañado cunha serie de parámetros co seguinte formato:

nome-evento '("*lista-de-argumentos*")' '['*condicion-de-garda*']' '/' *acción*
'^' *cláusula-envío*

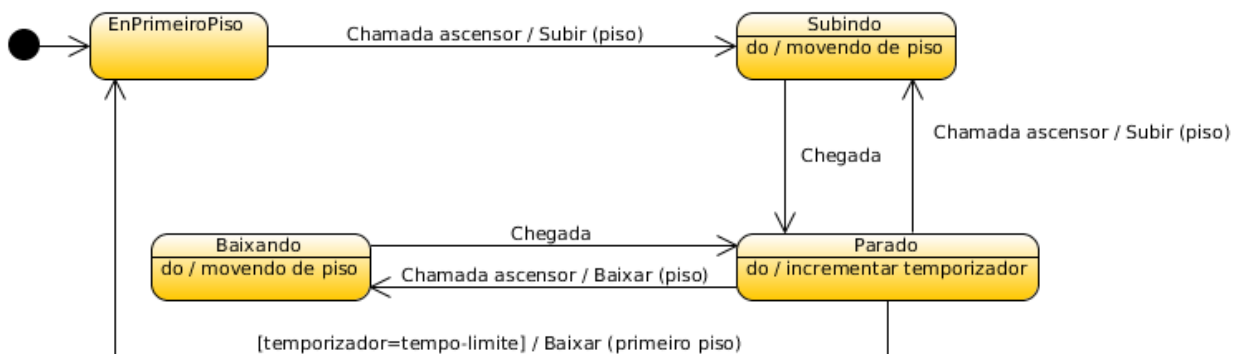
Onde:

- **nome-evento** e *lista-de-argumentos* describen o evento que dá lugar á transición e forman o que se denomina *event-signature*.
- **condicion-de-garda** é unha condición (expresión booleana) adicional ao evento e necesaria para que a transición aconteza.
Se a *condicion-de-garda* se combina cunha *event-signature*, entón para que a transición se dispare teñen que suceder dúas cousas: debe acontecer o evento e a condición booleana debe ser verdadeira.
- **acción** é unha acción que se executa cando se dispara a transición e que, por exemplo, pode ser unha chamada a unha operación.
É posible ter unha ou varias *acción* nunha transición de estado delimitadas co carácter "/".
- **cláusula-envío** é unha acción adicional que se executa co cambio de estado, por exemplo, o envío de eventos a outros paquetes ou clases.

Exemplo:



Imos ver un exemplo dun diagrama de estado para un ascensor



O ascensor empeza estando no primeiro piso. Pode subir ou baixar. Se o ascensor está parado nun piso, acontece un evento de tempo superado despois dun período de tempo e o ascensor baixa ao primeiro piso. Este diagrama de estado non ten un punto de finalización (estado final).

O evento da transición entre os estados **EnPrimeiroPiso** e **Subindo** ten un argumento, **piso**. O mesmo sucede cos eventos das transicións entre **Parado** e **Subindo** e entre **Parado** e **Baixando**. O estado **Parado** asigna o valor cero ao atributo temporizador. Despois incrementa continuamente o temporizador ata que aconteza o evento **Chamada ascensor** ou ata que a condición *[temporizador = tempo-límite]* se converta en verdadeira.

Outro exemplo, neste caso dos estados polos que pasa un teléfono nunha chamada telefónica:

