

Técnicas de modelado

Clase nro. 10
CURSO 2010

Problemas adicionales

Los ejemplos discutidos hasta ahora contienen simplificaciones que son difíciles de encontrar en sistemas de cierta complejidad.

Vimos p.ej. que

- cada actividad es realizada por una sola entidad,
- toda actividad se restringe mediante recursos y
- la única disciplina de cola presentada fue FIFO.

S.E.D. 2010

Problemas adicionales

Sin embargo las entidades pueden tener otros atributos que identifiquen sus individualidades,

así como pueden participar en varias actividades con otras entidades e independientes unas de otras.

Es decir que lo que definimos como **recursos pasivos** podrían ser reemplazados por **entidades activas**.

S.E.D. 2010

Problemas adicionales

Ejemplos:

- Los mecánicos en el taller de reparaciones pueden participar en otras actividades, por ejemplo tomar café, almorzar.
- Los pacientes en el hospital pueden tener otros atributos como edad, peso, etc.

También pueden existir **disciplinas de colas** basadas en los atributos de las entidades, (p.ej. el orden en que son operados los pacientes depende del tipo de operación requerida).

Luego de realizada una actividad, se puede **ramificar** la elección de la siguiente actividad, según diversos criterios.

S.E.D. 2010

Entidades cooperantes

Al estructurar una simulación debemos decidir qué objetos serán **entidades** y cuáles serán **recursos**.

Aquellos elementos que son todos idénticos, que son usados y liberados por otros objetos, se describen mediante **recursos**.

Deben ser descriptos como **entidades** aquellos objetos que:

- a) deben ser identificados individualmente o tienen atributos.
- b) son parte importante en otras actividades independientes y deben llevar el tiempo de la actividad en su reloj.

S.E.D. 2010

Entidades cooperantes

Cuando dos entidades dependen una de otra en el sentido de que participan de la misma actividad, se dice que son **cooperantes** en esa actividad.

Cada entidad tiene su ciclo o flujo propio de actividades, los cuales juntos completan todo el diagrama de actividades.

Ejemplo: mecánico y máquinas cooperan en las actividades **repair** y **remove**. Como el mecánico tendrá atributos propios, entonces puede ser asignado a distintas máquinas. Ver figura libro.

(cuidado: no hacerlo más complejo de lo necesario).

S.E.D. 2010

Entidades cooperantes

Cuando dos entidades cooperan en una actividad, ambas entidades deben ser agendadas desde el evento C correspondiente en el método de las tres fases;

En el método de las dos fases se verán distribuidos entre varios eventos fijos.

Pero surge un problema cuando finaliza una actividad cooperada, ya que cada entidad se debe agendar a dos eventos fijos independientes y cada actividad finaliza con un solo evento fijo y no dos.

S.E.D. 2010

Entidades cooperantes

Existen dos formas de resolver este problema:

- 1) Se agenda una de las entidades a un **evento ficticio fijo** (numerado con el 0) lo cual quiere decir que al final de esta actividad, la entidad se pierde y no puede ser puesta en cola o participar de otra actividad.
- 2) En caso de que ambas entidades deban estar accesibles, como ellas son **agendadas una detrás de la otra**, la última puede tomarse del calendario para ser tratada en lo que corresponda. Ejemplo: mecánico y máquinas accesibles para continuar c/u con su ciclo de vida.

S.E.D. 2010

Atributos de las entidades

Las distintas **características de las entidades** influyen en la duración y elección de actividades, por lo tanto son incluidas en el modelo como atributos de las entidades.

Los **atributos** pueden ser individuales a cada entidad o propios de una clase.

Un atributo puede ser una variable **entera, real** (peso, edad, altura), puede ser **enumerable** (color de ojos) o **contener datos complejos** (historia médica).

En una clase de entidades, cada entidad tendrá un valor para cada atributo, el cual estará fijo o cambiará durante la simulación.

S.E.D. 2010

Atributos de las entidades

Cuando existen atributos que diferencian a una clase de las otras, se pueden declarar estos atributos en un tipo record llamado **class_attribute** que luego se pueden usar referenciando a cada clase específica que corresponda.

Como se puede ver en la pag 137; si existen dos clases de máquinas con diferentes tiempos de reparación,

luego el número de clase de cada entidad puede ser utilizada para acceder a los valores de los atributos pertinentes declarados como **class_attribute**.

S.E.D. 2010

Atributos de las entidades

En Pascal, los tipos declarados como record no pueden ser cambiados en tiempo de ejecución, por ello se debe extender el tipo entidad con los atributos necesarios, así como **modificar** los procedimientos **new_entity** y **dis_entity**, para por último volver a **recompilar** el Pascal_SIM. EOSimulator permite extender la clase **Entity** mediante herencia, no requiere recompilar módulos existentes.

Cuando existe más de una clase de entidades y c/u de ellas con atributos propios, se pueden declarar un tipo record dinámico con diferentes campos según la clase que corresponda (ver libro pág. 136).

S.E.D. 2010

Atributos de las entidades

Las entidades pueden tener atributos de tipo booleano como ser mayor de 10 años, máquina rota, etc. Ello determina que un **conjunto de entidades posean esas propiedades**.

Esto hace que a veces, sea más eficiente utilizar la estructura *set* para manejarlas, en lugar de utilizar directamente el atributo booleano que le correspondería.

Consideremos el atributo booleano *máquina rota*. Se puede implementar una variable *set* de máquinas que han estado rotas; luego simplemente verificamos, mediante el atributo *attr*, si la máquina es parte o no de ese conjunto (pág. 138).

S.E.D. 2010

Prioridades y disciplinas de colas

Existen disciplinas de colas preestablecidas como ser: FIFO (first-in, first-out), LIFO (last-in, first-out) y SIRO (select in random order).

Se pueden implementar fácilmente con operaciones en la estructura de colas y agregando sorteo discreto del número de la entidad en la cola a ser procesada.

Se pueden definir disciplinas en las colas según los atributos de las entidades estableciendo prioridades.

En estos casos se deben realizar búsquedas en la cola según el atributo que determina la prioridad.

S.E.D. 2010

Prioridades y disciplinas de colas

Por ejemplo si le agregamos el atributo edad a las entidades pacientes del hospital y decimos que tendrán prioridad para la internación los más ancianos, deberemos buscar y retirar de la cola la entidad cuyo atributo edad sea el mayor.

Algunos lenguajes (GPSS) asignan prioridades como un atributo numérico de la entidad, entonces en los eventos condicionados se retira de la cola aquella entidad con prioridad de valor más alto, en caso de igualdad se usa una disciplina FIFO para decidir que entidad tomar.

EOSimulator: EntityQueue y clases derivadas.

S.E.D. 2010

Prioridades y disciplinas de colas

Las prioridades también pueden estar dadas por el **orden** en que se procesan los **eventos condicionados**, en el método de las tres fases, o en el orden en que se realizan las condiciones en el método a eventos.

Pueden surgir problemas cuando no existe independencia entre eventos condicionados, sino que un evento condicionado debe ser re-llamado desde otro o cuando hay mutua dependencia entre ellos. En estos casos conviene tener un solo evento condicionado y no dos o más por separado (ver pag 141).

S.E.D. 2010

Ramificaciones

Decisiones en distintas etapas de la simulación se pueden tomar según distintos criterios:

- 1) estocásticamente o mediante
- 2) elección según atributos.

Ramificación **estocástica** se realiza sorteando según una distribución de probabilidades (ver libro, pág, 142).

Ramificación **por atributos** se realizan testeando los valores apropiados para decidir qué rama tomar. La decisión se toma en los eventos fijos que representan el fin de la actividad desde la cual la entidad se ramificará.

S.E.D. 2010

Resumen

La técnica de modelar atributos, entidades cooperantes, prioridades y ramificaciones es muy importante en la simulación.

Para manejar diversos *atributos* se debe redefinir el tipo entidad, los procedimientos de crear y eliminar entidades, así como recompilar Pascal_SIM, para cada simulación que así lo requiera.

S.E.D. 2010

Resumen

Prioridades se pueden determinar mediante 1) manejo del método de selección en las colas, 2) las características propias de las entidades y 3) mediante el orden de realización de las actividades. Estos tres criterios son intercambiables, aunque algunos lenguajes permiten nada más que uno de ellos.

La propia lógica y variabilidad del sistema determinará el uso de *ramificación* entre actividades.
Los criterios y método a elegir siempre deben ser aquellos que más se adapten al sistema en estudio, tratando de que el elegido sea el más flexible posible.

S.E.D. 2010