

Ingeniería de Software II

Diagrama de actividad



Diagrama de Actividad

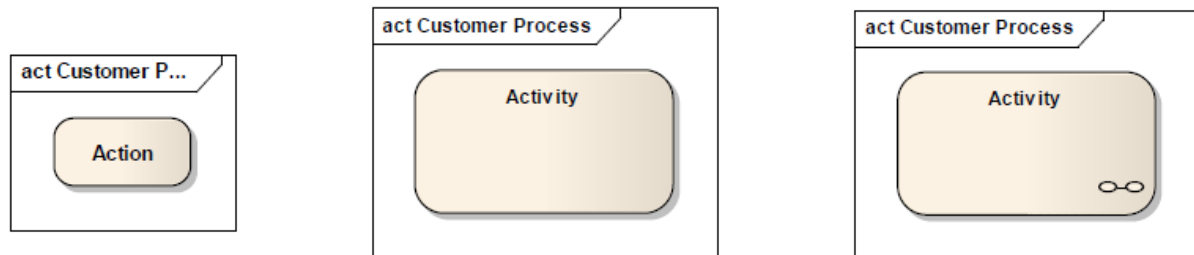
- ▶ Diagramas de actividad (DA) son como diagramas de flujo Orientado a Objetos (OO).
- ▶ Permiten modelar un proceso como una actividad que se compone de una colección de nodos conectados por arcos.
- ▶ Permiten:
 - Modelar flujo de trabajo
 - Modelar procesos de negocio
 - Pueden acompañar a cualquier elemento de modelado para expresar su comportamiento:
 - Casos de uso
 - Clases
 - Interfaces
 - Colaboraciones
 - Componentes

DA

- ▶ Red de nodos conectados por arcos.
- ▶ Existen tres categorías de nodos:
 - Nodo de acción:
 - representa unidades de trabajo que son atómicas dentro de la actividad.
 - Nodo de control:
 - controla el flujo por medio de la actividad.
 - Nodo de objetos:
 - representan objetos utilizados en la actividad.
- ▶ Los arcos (flujos de control o edges) representan flujo a través de la actividad.
 - Flujos de control
 - Flujos de objeto

Actividad y acción

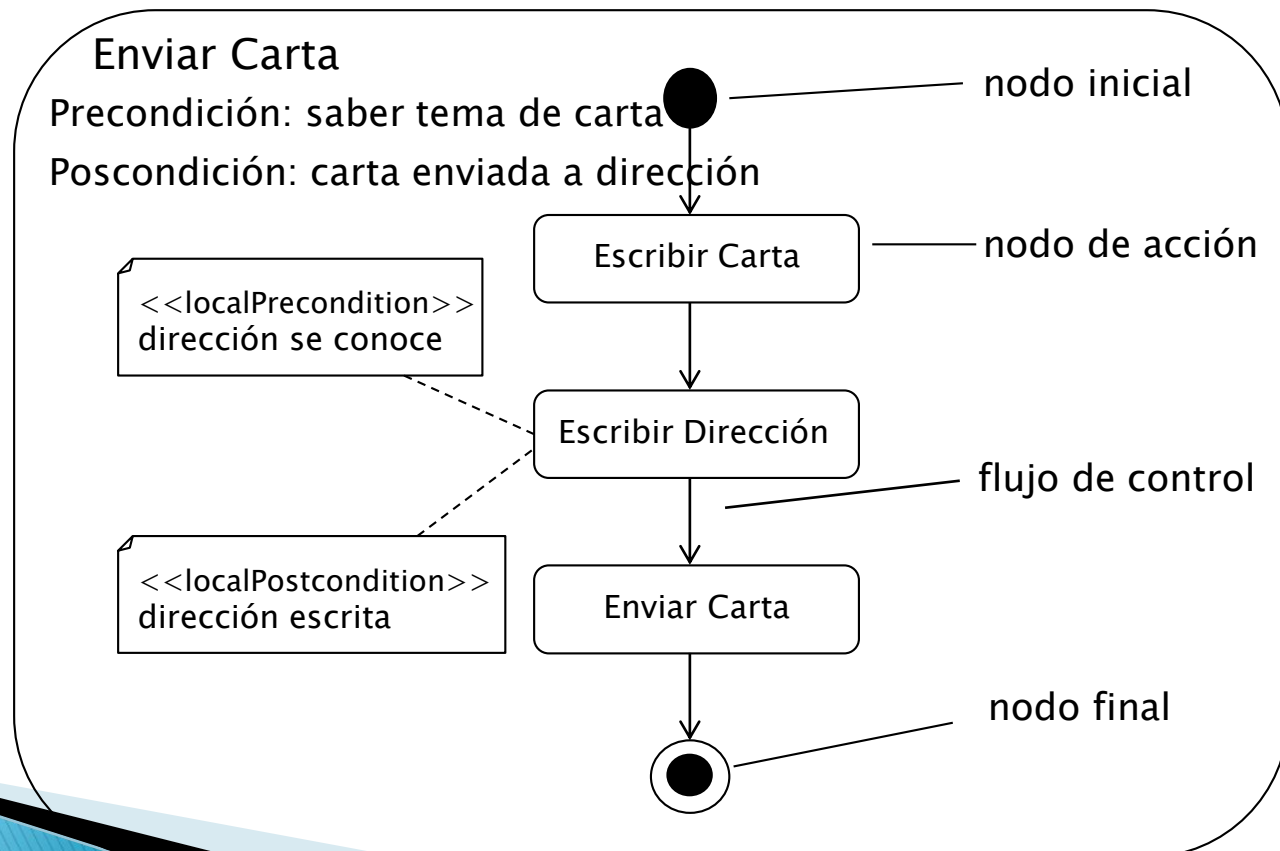
- ▶ La representación de Actividad y Acción:
 - Rectángulo con las puntas redondeadas.
 - La forma de expresión no queda impuesta por UML, se podría utilizar lenguaje natural, una especificación formal de expresiones, un metalenguaje, etc.
 - Una acción es atómica.
 - Su ejecución es instantánea y no puede ser interrumpida.



Actividad

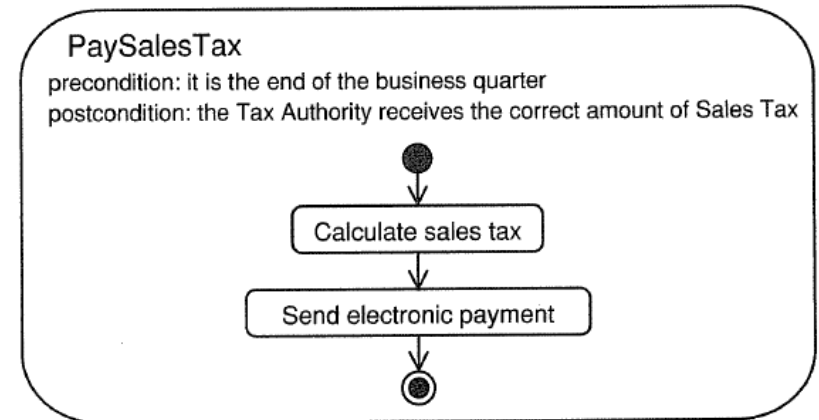
Las actividades pueden tener precondiciones y pos condiciones como los casos de uso:

Son redes de
nodos conectados
por flujo de control



CU vs DA

Use case: PaySalesTax	
ID: 1	
Brief description: Pay Sales Tax to the Tax Authority at the end of the business quarter.	
Primary actors: Time	
Secondary actors: TaxAuthority	
Preconditions: 1. It is the end of the business quarter.	
Main flow: 1. The use case starts when it is the end of the business quarter. 2. The system determines the amount of Sales Tax owed to the Tax Authority. 3. The system sends an electronic payment to the Tax Authority.	
Postconditions: 1. The Tax Authority receives the correct amount of Sales Tax.	
Alternative flows: None.	



CU expresa el comportamiento del sistema como una interacción entre un actor y el sistema

DA expresa el comportamiento del sistema como una serie de acciones.

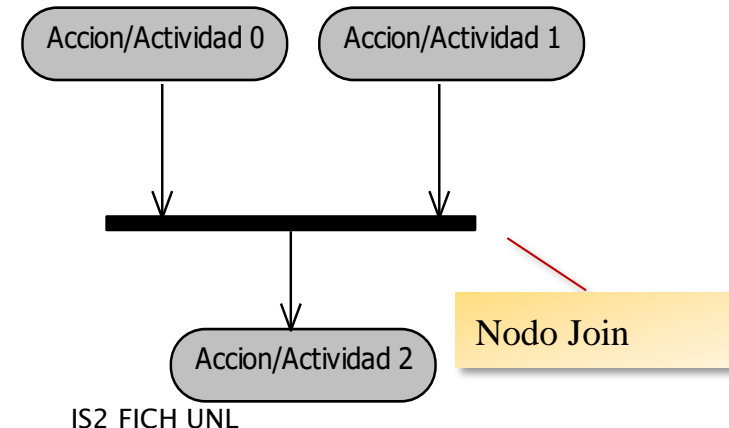
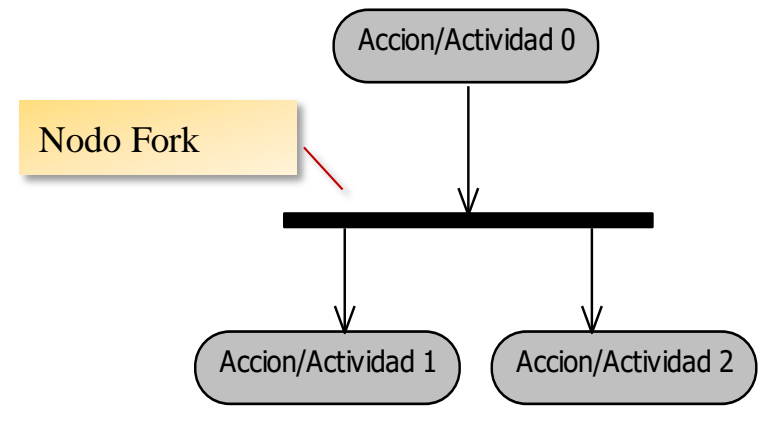
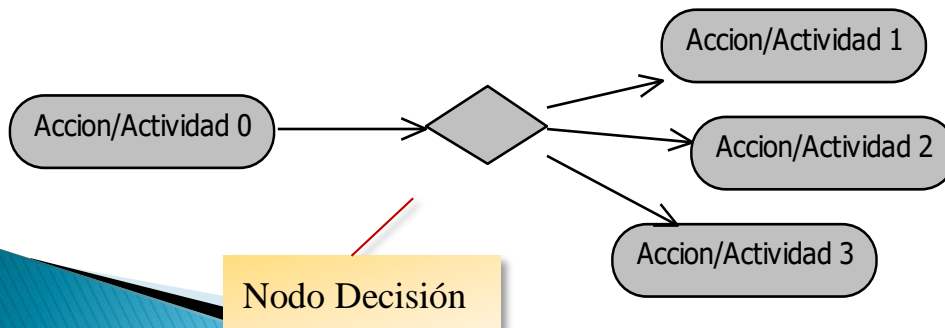
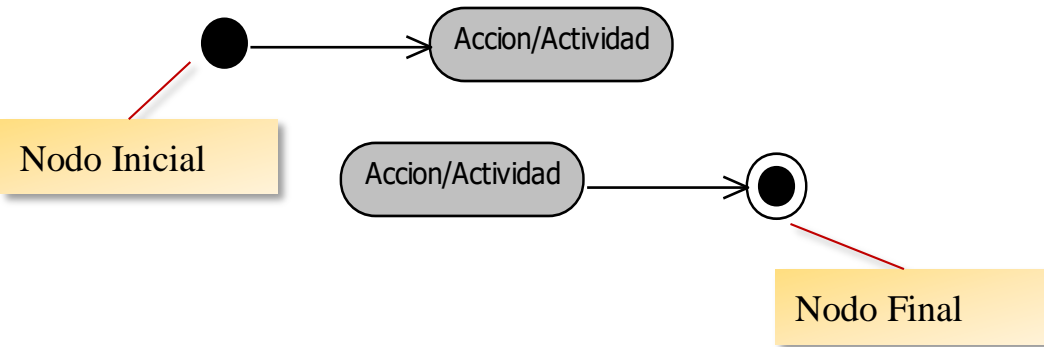
Son vistas complementarias del mismo comportamiento

Actividad y acción

- ▶ Una actividad puede descomponerse en más subactividades
 - Se representan a través de otros diagramas de actividades.
- ▶ Pueden ser interrumpidas.
- ▶ Tardan un cierto tiempo en completarse.
- ▶ En las actividades podemos encontrar otros elementos adicionales:
 - acciones de entrada (entry) y de salida (exit) del estado en cuestión
 - definición de submáquinas

Nodos de Control

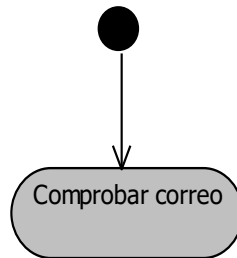
- ▶ Gestionan el flujo de control dentro de una actividad.



Nodo Inicial/Nodo final

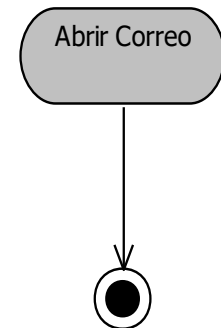
► Nodo inicial:

- Indica dónde empieza el flujo cuando se invoca una actividad
- Estado inicial



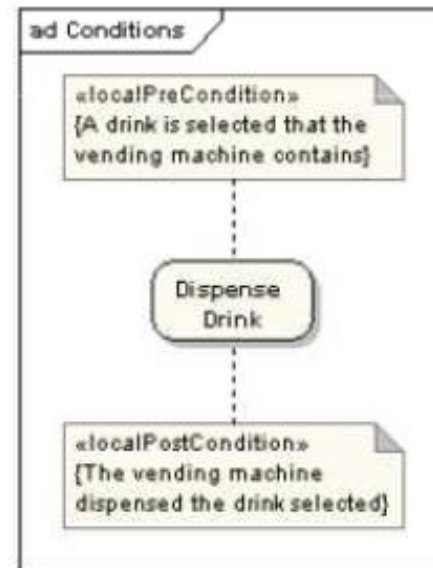
► Nodo final:

- Termina una actividad
- Detiene todos los flujos dentro de una actividad
- Estado de parada



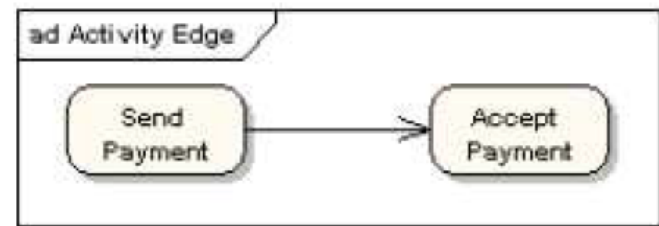
Restricciones de Acción

- Las restricciones se pueden adjuntar a una acción.
 - Precondiciones locales
 - Post condiciones locales.



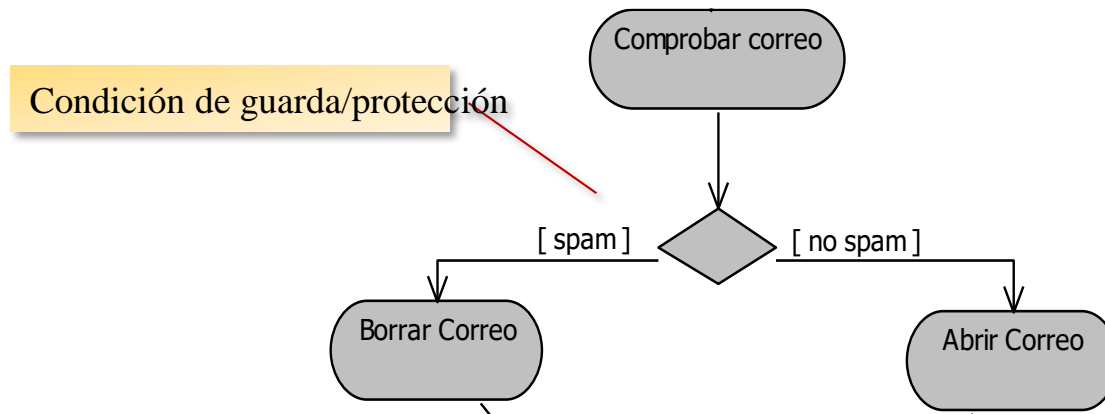
Transiciones o flujos de control

- ▶ Las transiciones reflejan el paso de un estado a otro, bien sea de actividad o de acción.
- ▶ Se produce como resultado de la finalización del estado del que parte el arco dirigido que marca la transición.



Bifurcaciones

- ▶ **Nodo decisión:** Posee un extremo de entrada y dos o mas de salida
 - Un token que llega al extremo de entrada se ofrecerá a todos los extremos de salida, atravesando uno.
 - Cada uno de los extremos → condición de protección (guarda), que debe evaluar a verdadera.



Bifurcaciones

- ▶ Un flujo de control no tiene porqué ser siempre secuencial, puede presentar caminos alternativos.
- ▶ Se utiliza como símbolo el rombo.
- ▶ En cada transición de salida se colocará una expresión que será evaluada una vez al llegar a la bifurcación.
- ▶ Las guardas de la bifurcación son excluyentes y contemplan todos los casos ya que de otro modo la ejecución del flujo de control quedaría interrumpida.

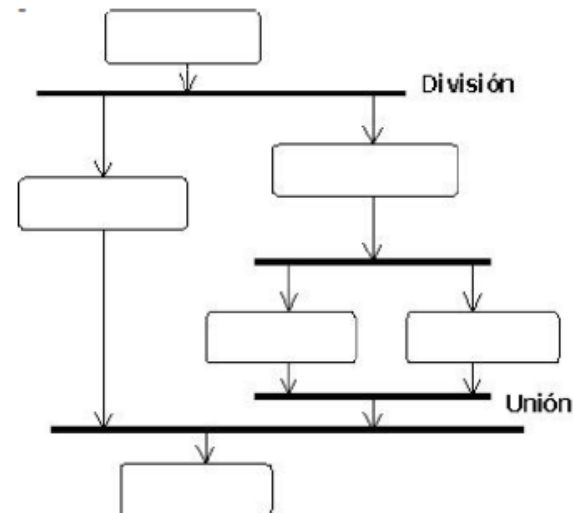
Bifurcaciones

- ▶ Para poder cubrir todas las posibilidades se puede utilizar la palabra ELSE, para indicar una transición obligada a un determinado estado cuando el resto de guardas han fallado.



División y unión

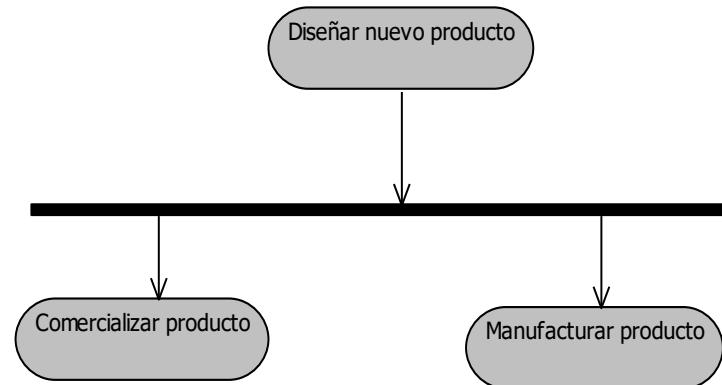
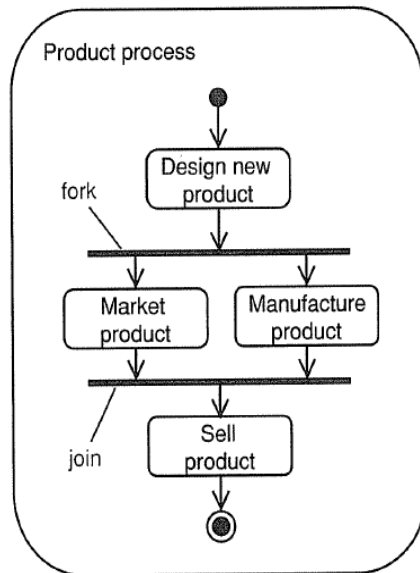
- ▶ Hay casos en los que se requieren tareas concurrentes.
 - UML representa gráficamente con una línea ancha:
 - el proceso de división que representa la concurrencia
 - el momento de la unión de nuevo al flujo de control secuencial



División y unión

► Nodo fork

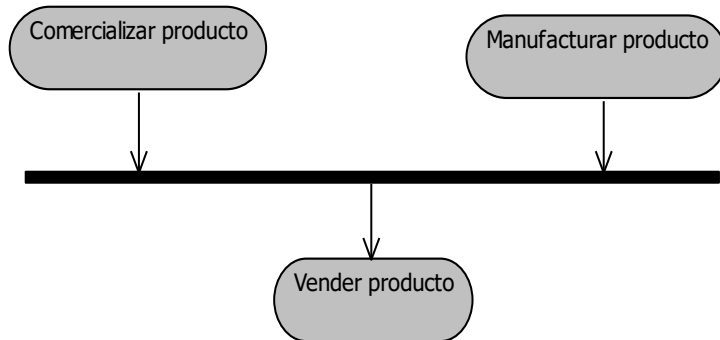
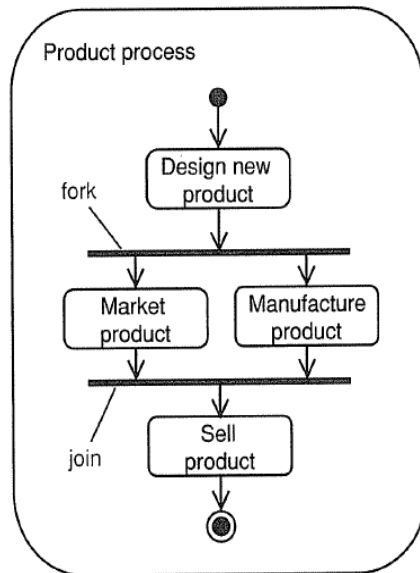
- Divide el flujo en múltiples flujos concurrentes
- Tiene un extremo de entrada y dos o mas de salida paralelos



División y unión

► Nodo Join

- Múltiples extremos entrantes y uno solo de salida
- Sincroniza flujos.
 - Tiene un único token en el extremo de salida



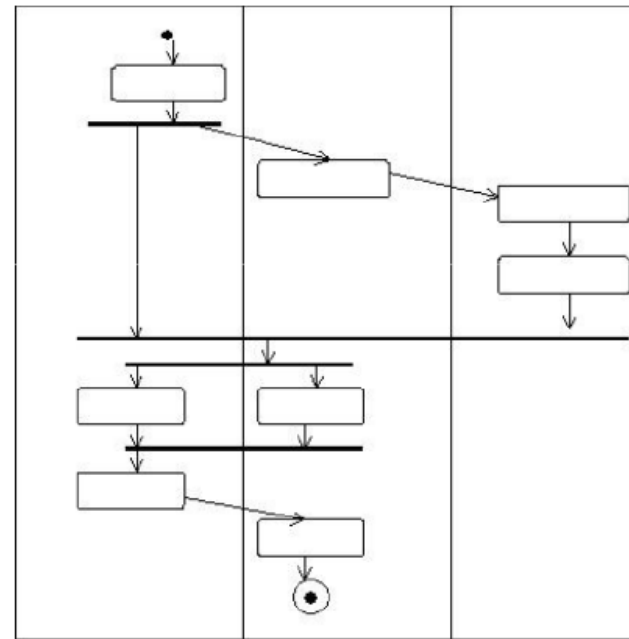
Particiones

- ▶ Para lograr diagramas de actividad más fáciles de leer
- ▶ Se pueden dividir las actividades en particiones utilizando líneas verticales, horizontales o curvas.
- ▶ Cada partición representa un agrupamiento de más alto nivel de acciones relacionadas.
- ▶ Se pueden llamar swimlines

Particiones

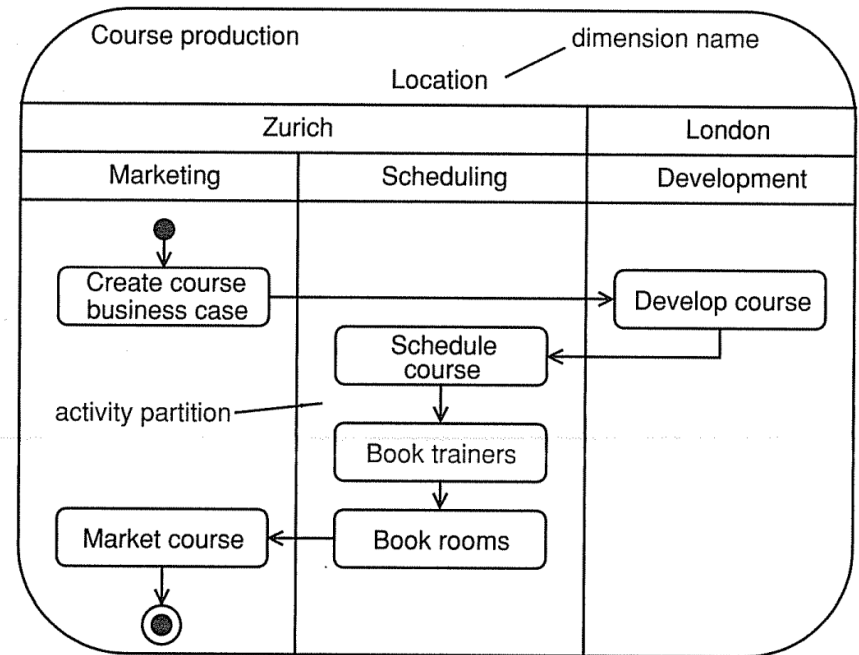
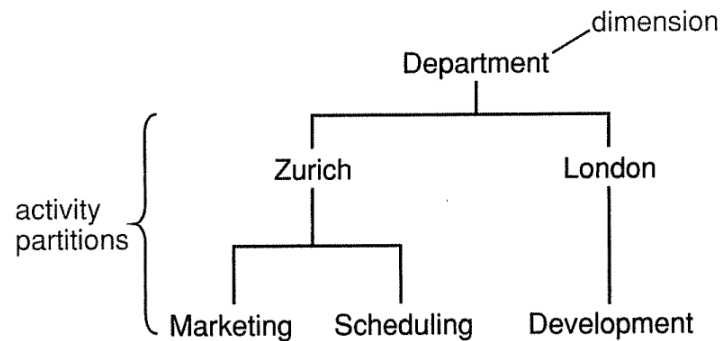
Cuando se modelan flujos de trabajo de organizaciones, es especialmente útil dividir los estados de actividades en grupos, cada grupo tiene un nombre concreto y se denominan calles.

Cada calle representa a la parte de la organización responsable de las actividades que aparecen en esa calle.



Particiones

Zuhike Engineering AG

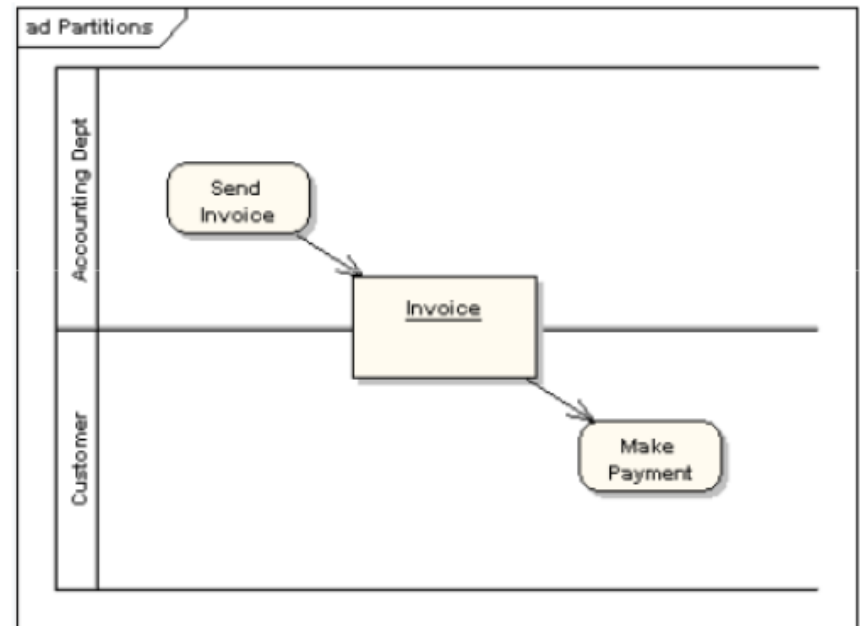


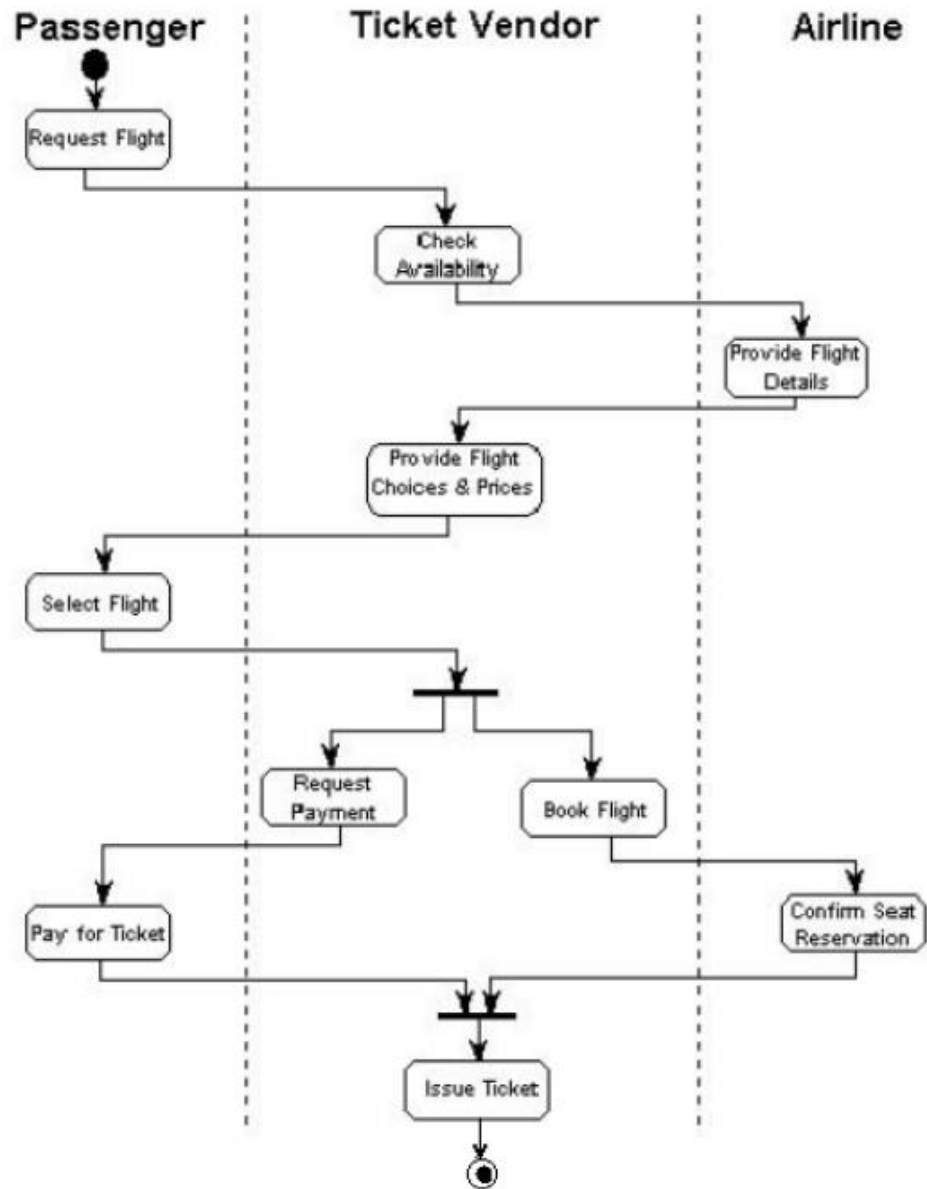
Particiones

- ▶ Las calles pueden ser verticales u horizontales.



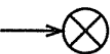
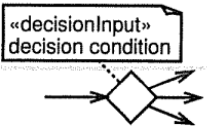


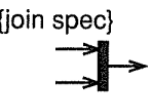
Algunas veces no es posible organizar los nodos en particiones verticales u horizontales sin dificultar la lectura del diagrama.

Si una acción reside en más de una partición, se lista cada nombre de ruta de acceso de partición separado por comas.



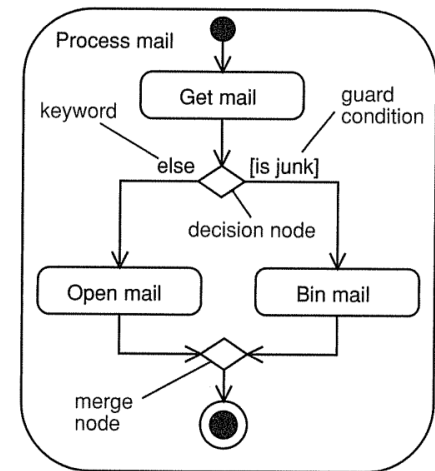


Nodos de control

Syntax	Name	Semantics	Section
	Initial node	Indicates where the flow starts when an activity is invoked	14.8.1
	Activity final node	Terminates an activity	Final nodes 14.8.1
	Flow final node	Terminates a specific flow within an activity – the other flows are unaffected	
	Decision node	The output edge whose guard condition is true is traversed May optionally have a «decisionInput»	14.8.2
	Merge node	Copies input tokens to its single output edge	14.8.2
	Fork node	Splits the flow into multiple concurrent flows	14.8.3
	Join node	Synchronizes multiple concurrent flows May optionally have a join specification to modify its semantics	14.8.3

Nodo merge

- ▶ Posee dos o más arcos de entrada y un único arco de salida.
- ▶ Fusionan los flujos de entrada en un flujo de salida simple.
- ▶ Semántica: Todos los tokens ofrecidos en los arcos de entrada son ofrecidos en el arco de salida y no hay modificación en el flujo o en los tokens.



Semántica del DA: Juego del token

DA modelan el comportamiento al utilizar el juego token.

Este juego describe el flujo de tokens alrededor de una red de nodos y arcos de acuerdo a reglas específicas.

Los token en los DA pueden representar:

- Flujo de control
- Un objeto
- Datos

El estado del sistema en cualquier punto en el tiempo está determinado por la posición de sus tokens

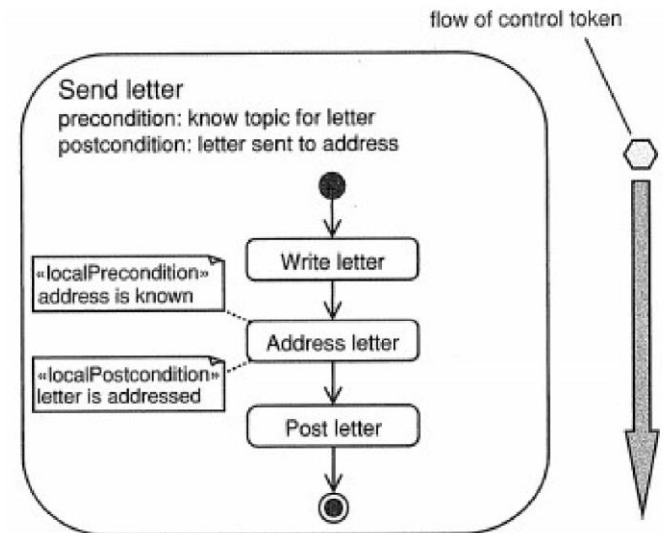
Semántica del DA: Juego del token

- ▶ Los tokens se pasan de un nodo origen a un nodo destino a través de un arco.
- ▶ El movimiento de un token está sujeto a condiciones
 - Sólo puede ocurrir cuando todas las condiciones se cumplen.

Las condiciones van variando dependiendo del nodo.

Para los nodos acción las condiciones son:

- Las postcondiciones del nodo origen
- Las condiciones de guarda del flujo de control (edge/flecha)
- Las precondiciones del nodo destino



Semántica del DA: Juego del token

► Nodo acción

- Contiene detalles de la acción
- Se ejecutan cuando:
 - Existe un token en cada uno de los extremos de entrada AND
 - Token de entrada satisfacen las precondiciones del nodo de acción.
 - El nodo no está listo para ejecutarse hasta que esten los tokens en cada uno de sus extremos.

