

# TEMA 8. DIAGRAMAS DE ACTIVIDAD

## ANÁLISIS Y DISEÑO DE SISTEMAS DE INFORMACIÓN



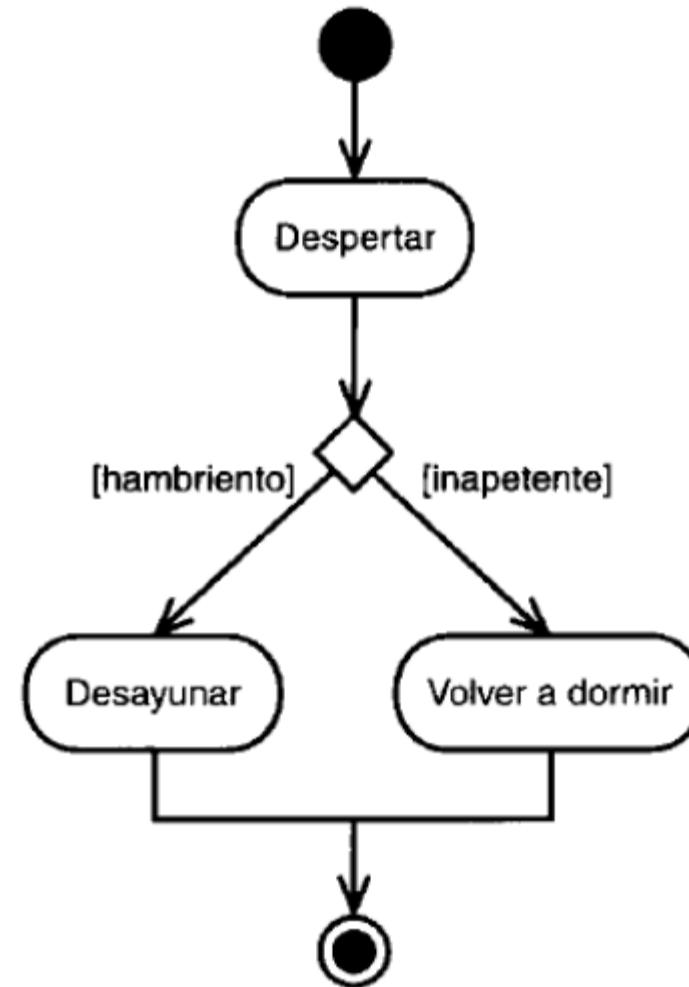
Universidad  
Francisco de Vitoria  
**UFV** Madrid

# 1.INTRODUCCIÓN

Enfoque del diagrama de actividad:  
definición de procedimientos

Orientados a flujos

Se utiliza para definir: el  
comportamiento de un método,  
definir estados complejos, y sobre  
todo para definir el  
**comportamiento de los casos de  
uso** y para **modelar procesos de  
negocio**.



## 2. ACTIVIDAD

Especifica del comportamiento definido por el usuario en diferentes niveles de granularidad

Ejemplos:

- Definición del comportamiento de una operación en forma de instrucciones individuales.
- Modelar el curso de las acciones de un caso de uso
- Diseñar las funciones de un proceso empresarial

Una actividad es un gráfico que representa una dirección

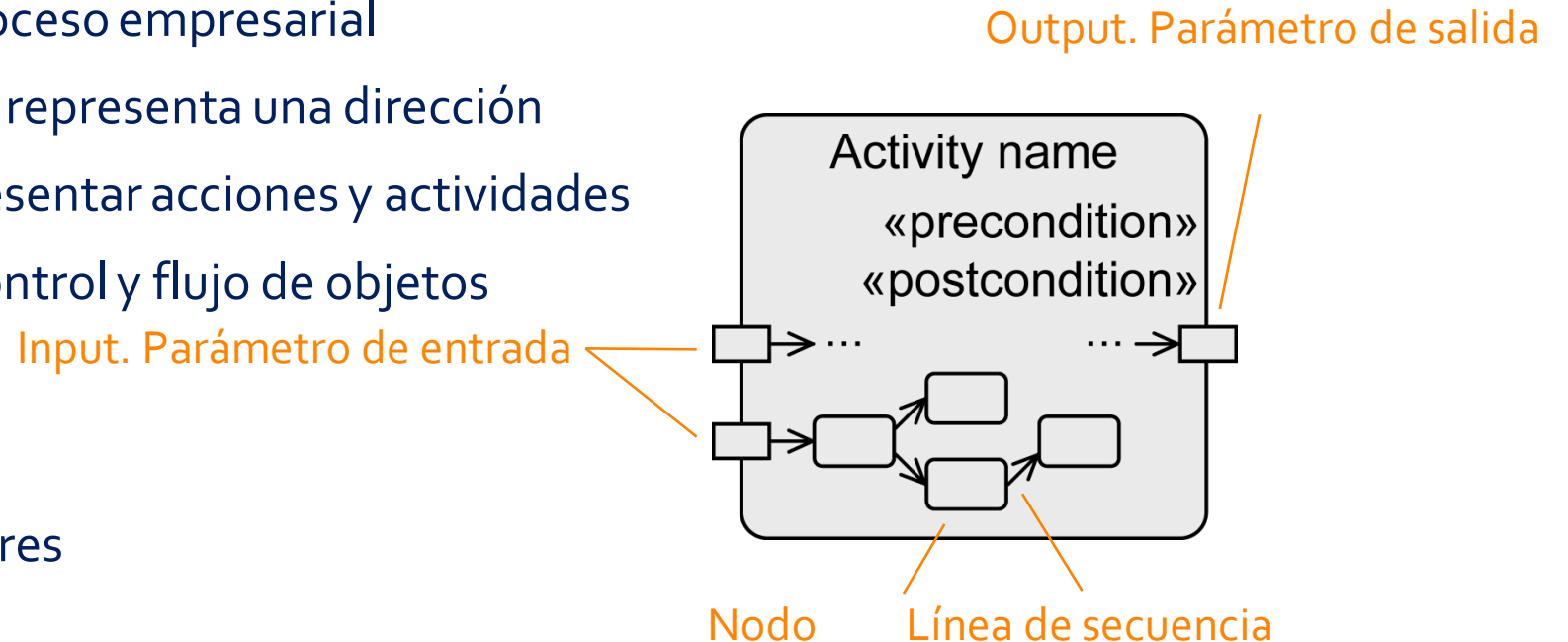
Los nodos se utilizan para representar acciones y actividades

Las líneas de secuencia: para control y flujo de objetos

Opcional:

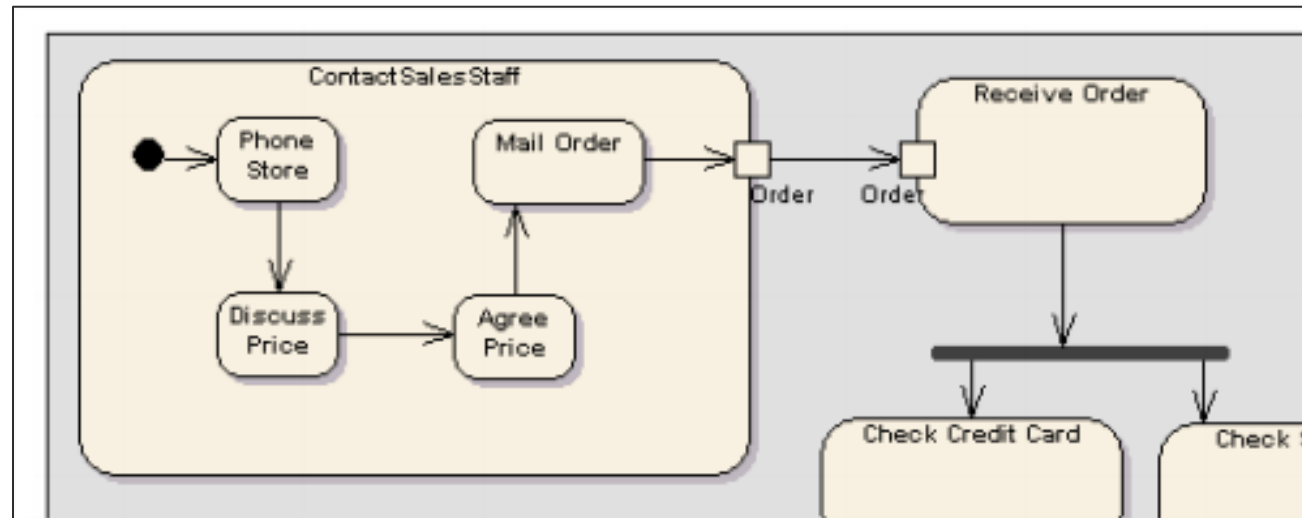
- Parámetros
- Condiciones previas y posteriores

Activity A



## 2. ACTIVIDAD

### Activity A



# 3. ACCIÓN

**Es el elemento básico para especificar el comportamiento definido por el usuario**

Action A

Puede finalizarse en cualquier momento.

No hay reglas específicas para la definición de una acción.

→ Definición en lenguaje natural o en cualquier lenguaje de programación

Procesar valores de entrada para producir valores de salida

Notación especial para tipos de acciones predefinidos. Sobre todo:

- Acciones basadas en eventos
- Acciones de comportamiento de llamadas

# 4. LÍNEAS DE SECUENCIA O DE FLUJO

Conectar actividades y acciones entre sí

Expresar el orden de ejecución

Tipos

Definición de control de flujo

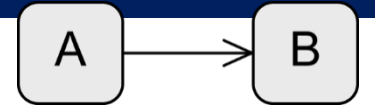
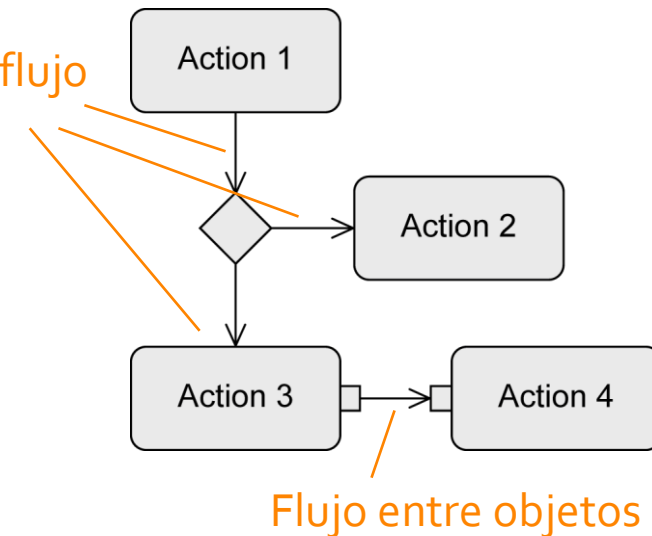
- Orden entre nodos

Definición el flujo entre objetos

- Se utiliza para intercambiar datos u objetos.

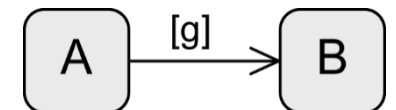
- Expresa una dependencia de datos entre nodos

Líneas de control del flujo



Guarda (condición)

El control y el flujo de objetos solo se da si las condiciones entre corchetes se evalúan como verdaderas



# 5.ACTIVIDADES DE INICIO Y FIN



## Nodo inicial

Inicia la ejecución de una actividad.

Todos los flujos parten de este elemento.

Mantiene los flujos hasta que los sucesivos nodos los acepten



## Nodo final de actividad

Finaliza todos los flujos de una actividad

El primer flujo que llega al nodo final de la actividad finaliza toda la actividad

## Excepción:

Nodo final de flujo



Finaliza un camino de ejecución de una actividad

Todos los demás flujos de la actividad no se ven afectados

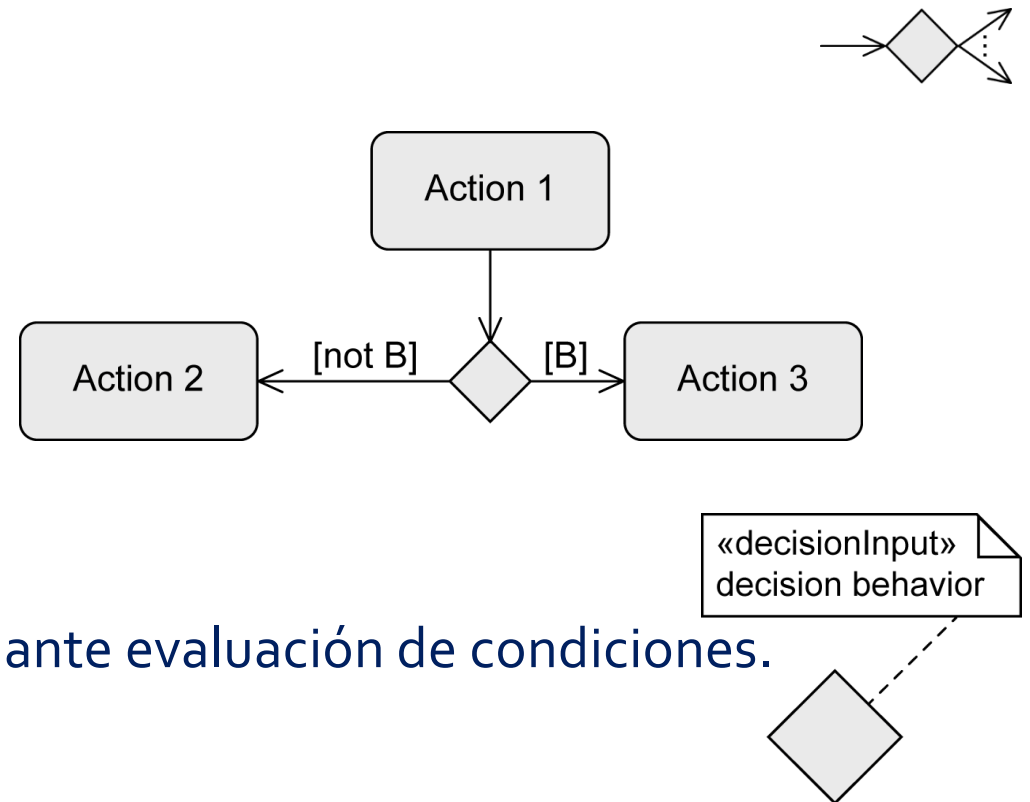
# 6. FLUJOS ALTERNATIVOS

Para definir ramas alternativas

Los flujos salientes tienen condiciones

Sintaxis: [expresión booleana]

Las condiciones deben ser mutuamente excluyentes



Es un comportamiento de decisión

Especificar el comportamiento que es necesario mediante evaluación de condiciones.

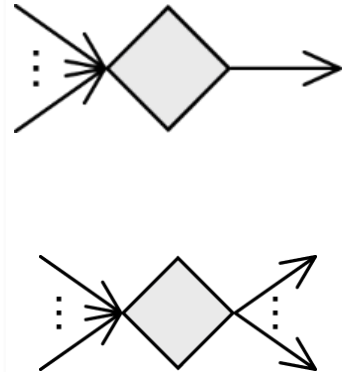
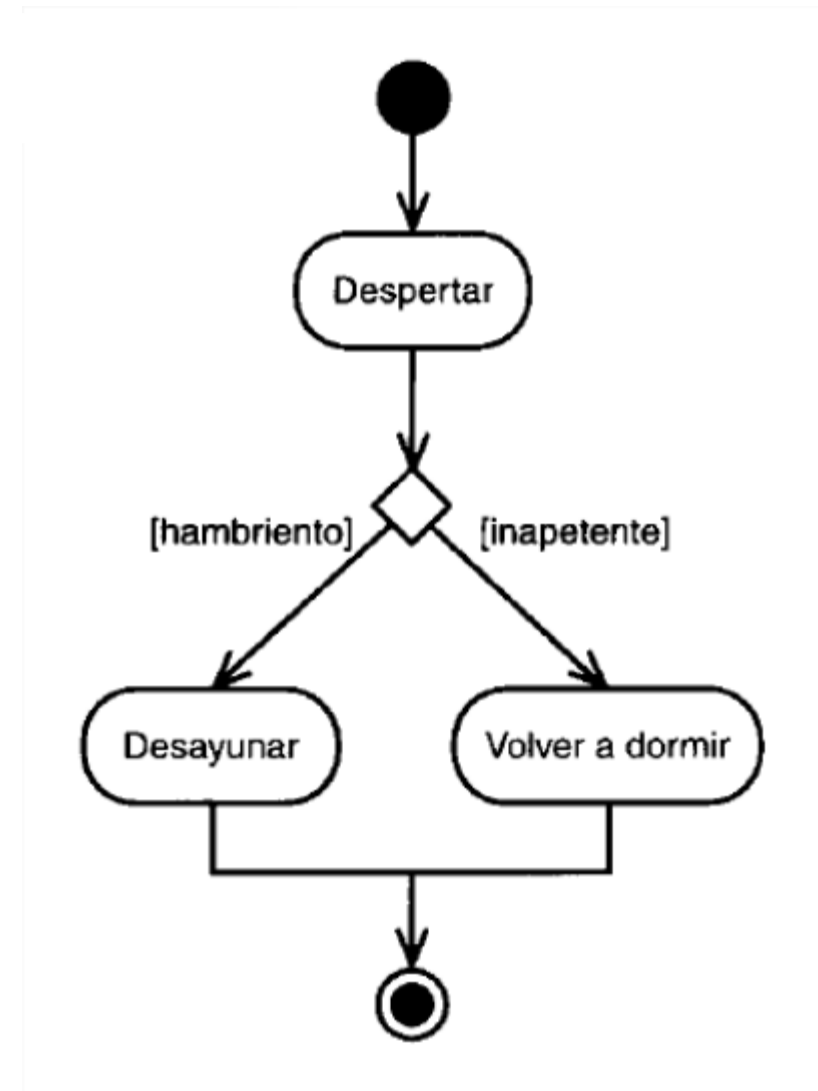


# ALTERNATIVE PATHS – MERGE NODE

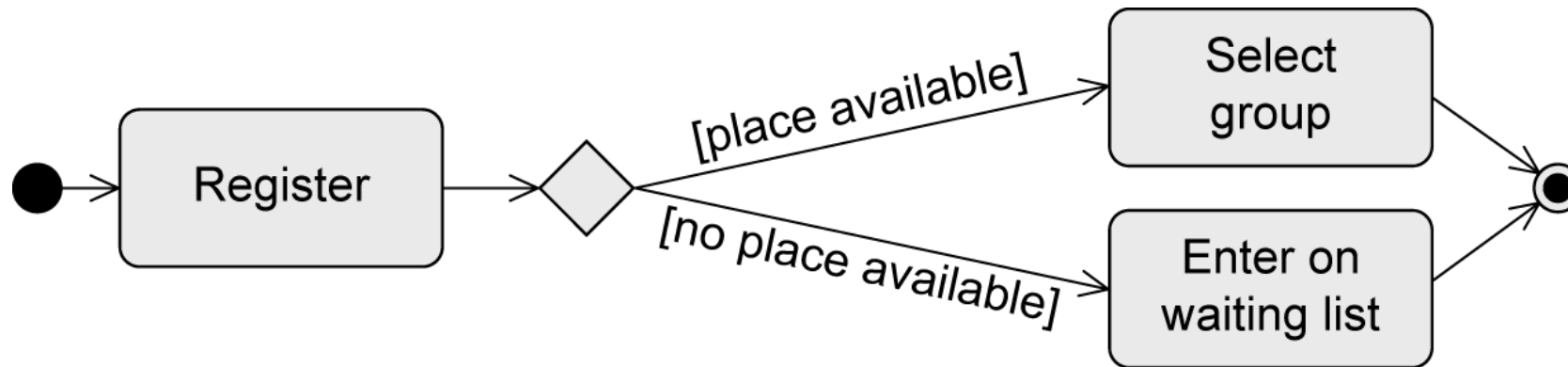
Para unir flujos o caminos alternativos  
Pasa el flujo al siguiente nodo

Nodo combinado de decisión y fusión:

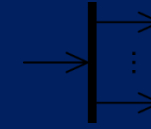
Los nodos de decisión y fusión también se pueden usar para modelar bucles:



# EJEMPLO

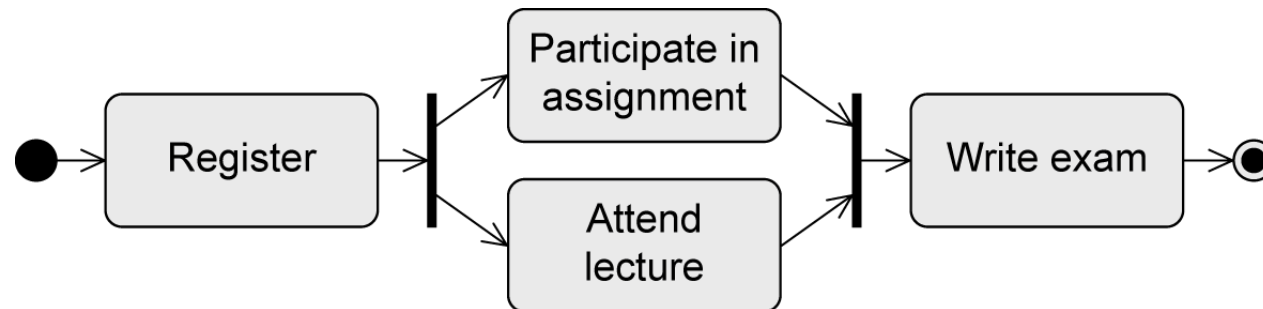


# RUTAS CONCURRENTES: NODO DE PARALELIZACIÓN



Para dividir la ruta en subrutas concurrentes

Ejemplo:



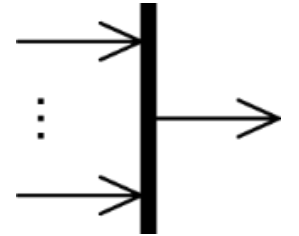
# RUTAS CONCURRENTES: NODO DE SINCRONIZACIÓN

Para fusionar subrutinas concurrentes

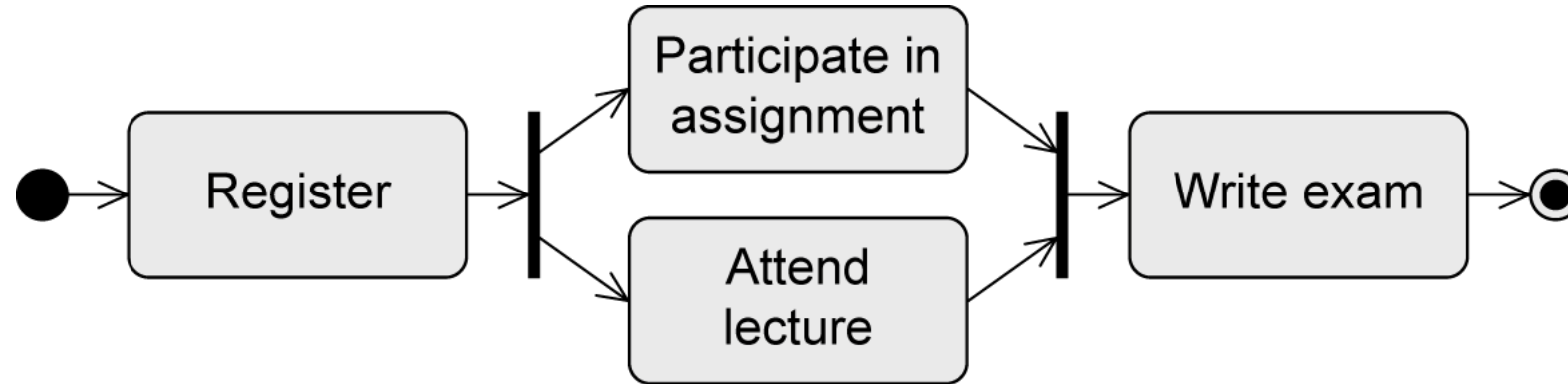
Espera hasta que todos los flujos entrantes estén presentes

Fusiona todos los flujos de control en uno solo y lo pasa

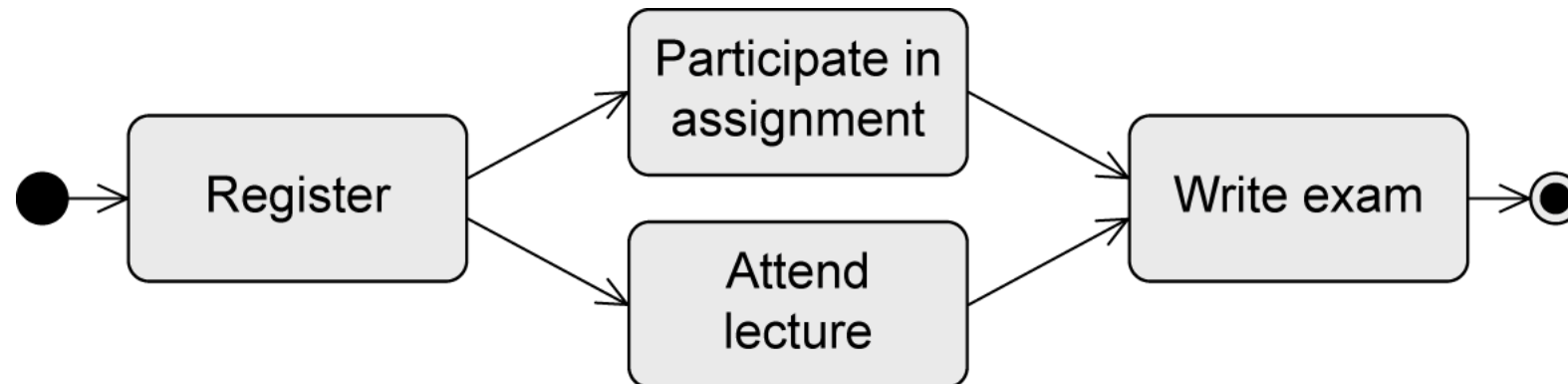
Pasa todos los flujos de objeto



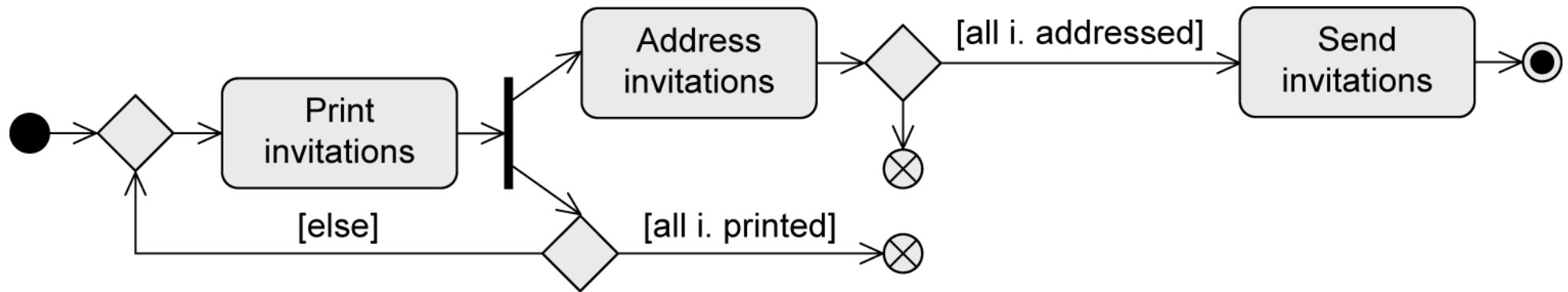
# EJEMPLO



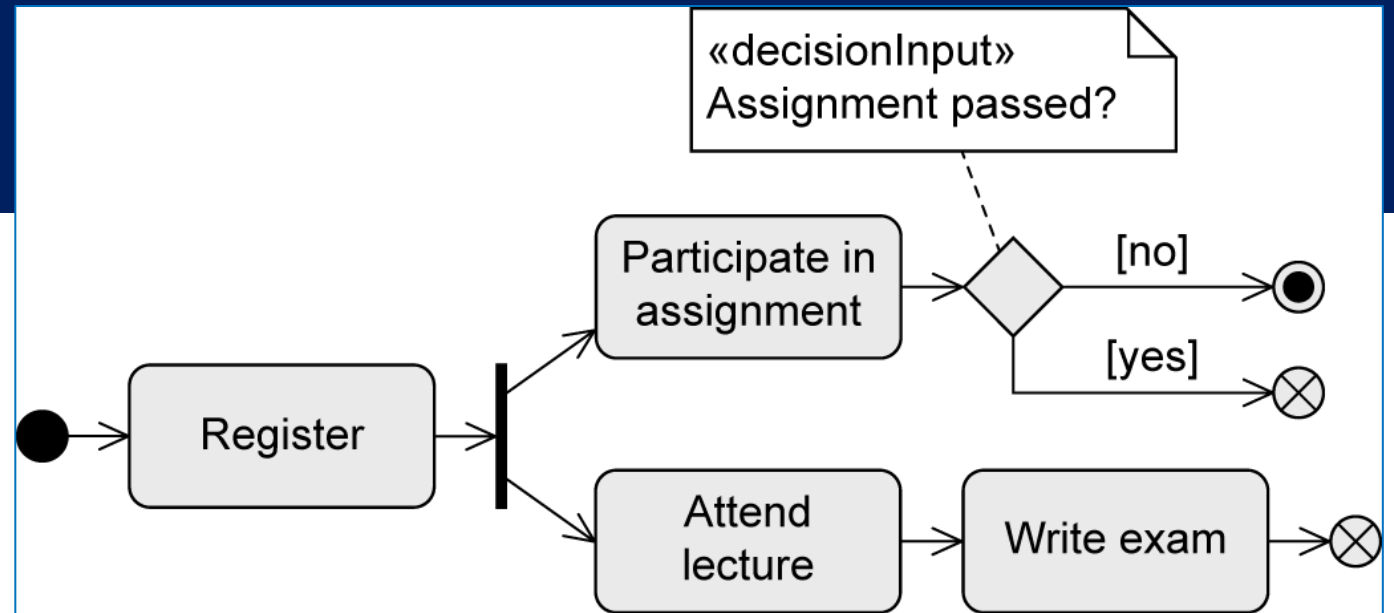
... equivalente a ...



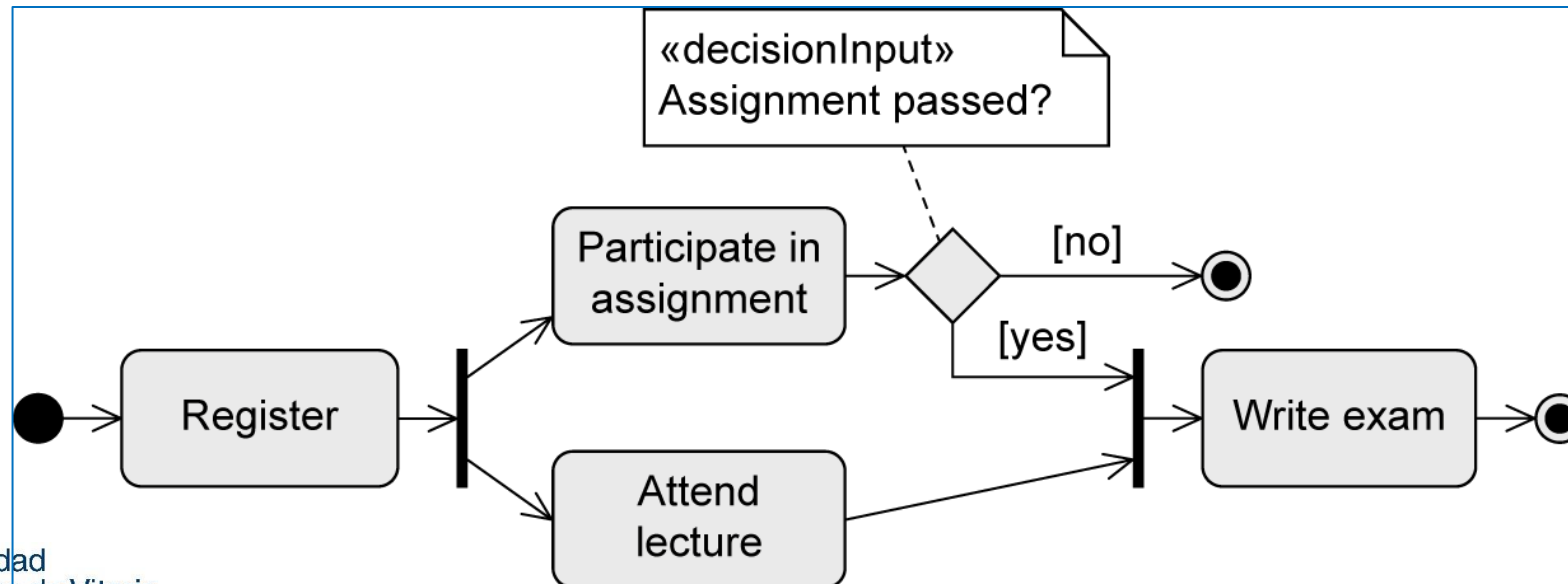
# EJEMPLO. CREARY ENVIAR INVITACIONES



# EJEMPLO



NO ES EQUIVALENTE... ¿POR QUÉ?



# NODO OBJETO

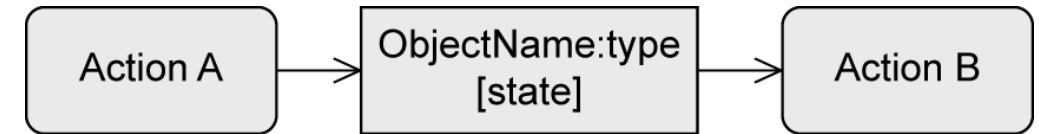
Object

Contiene objetos

Representa el intercambio de datos / objetos

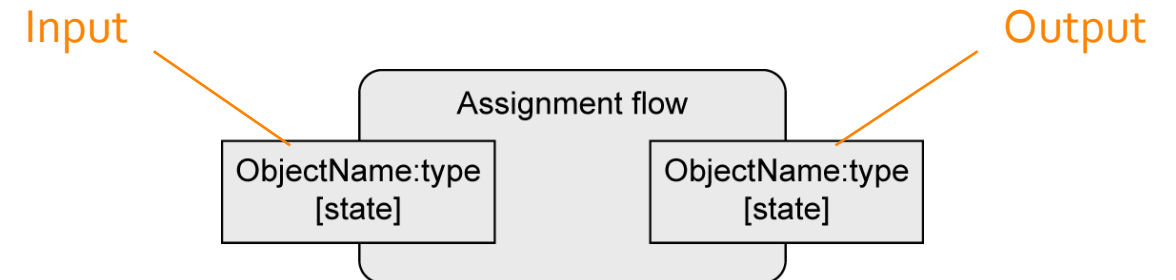
Es el origen y el destino de un borde de flujo de objetos

Información opcional: tipo, estado

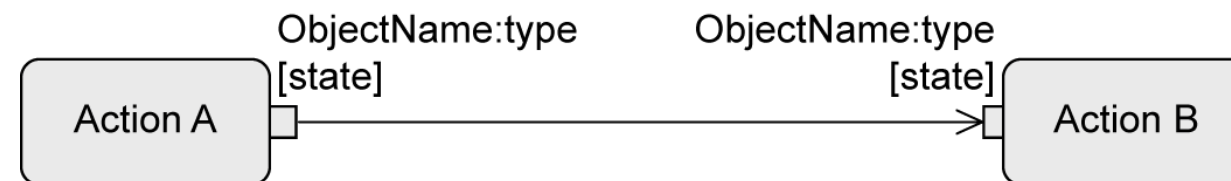


Variante de notación: nodo de objeto como parámetro

Para actividades

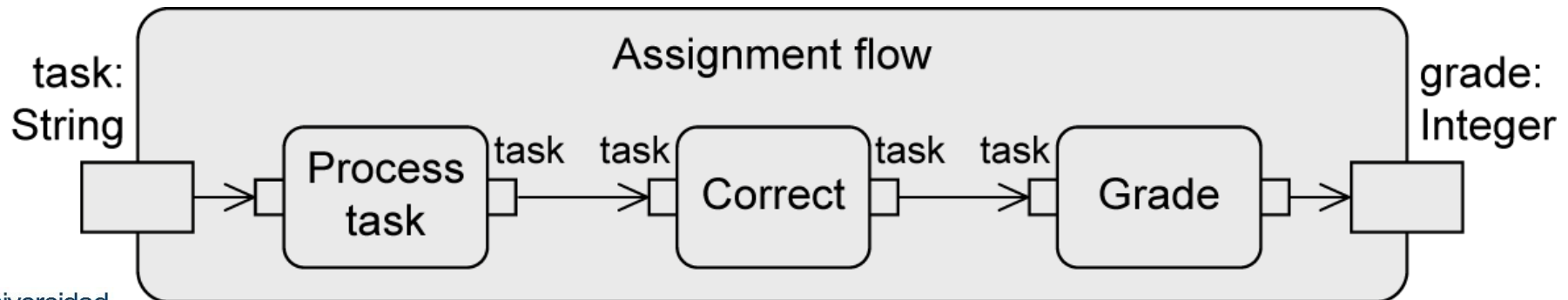
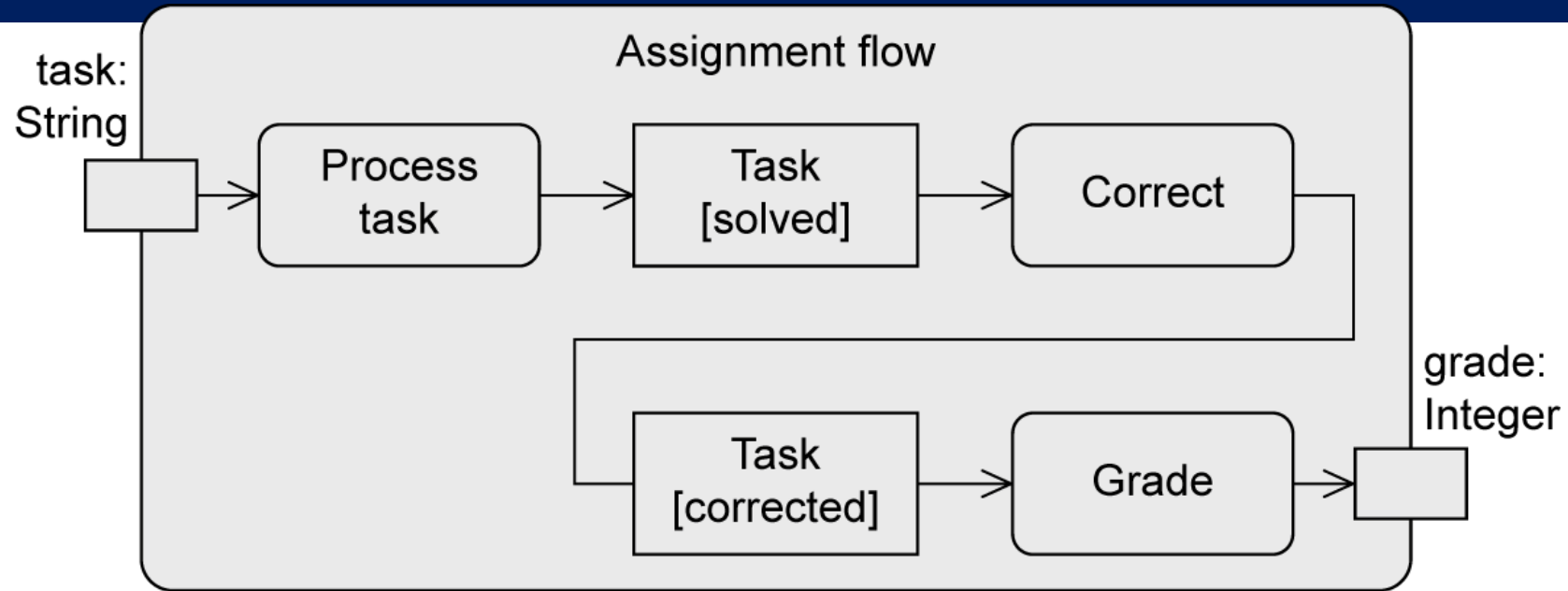


Para acciones





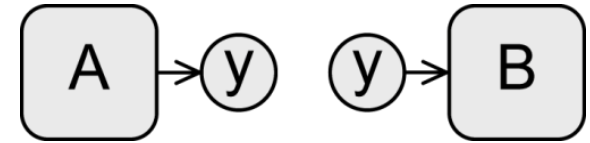
# EJEMPLO



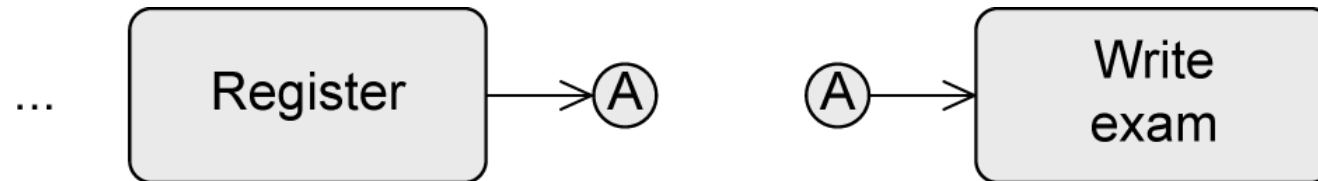
# CONECTOR

Se usa si dos acciones consecutivas están muy separadas en el diagrama.

Sin conector:



Con conector



# ACCIÓN DE COMPORTAMIENTO DE LLAMADA

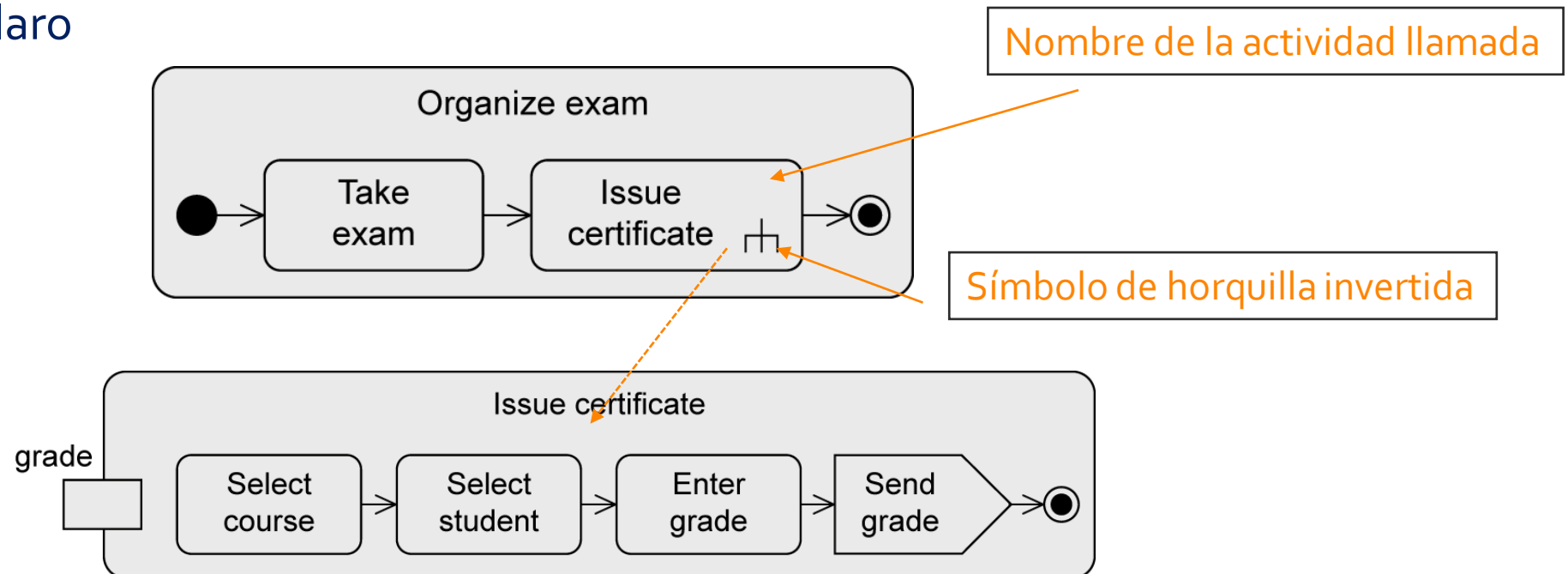
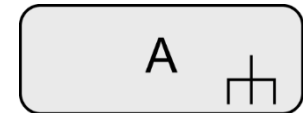
La ejecución de una acción puede llamar a una actividad.

El contenido de la actividad llamada se puede modelar en otro lugar.

Ventajas:

El modelo es más claro

Reutilización



# PARTICIÓN

Es similar "Carril"

Gráficamente o textual

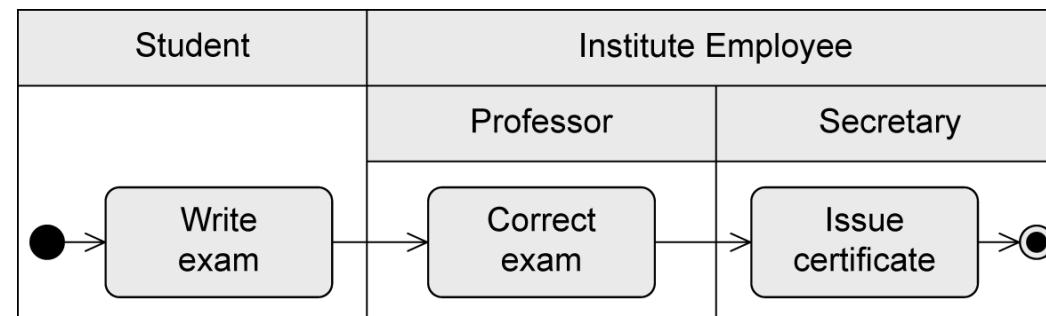
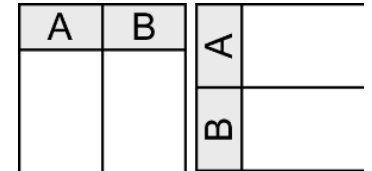
Permite la agrupación de nodos y bordes de una actividad por responsabilidades

Las responsabilidades reflejan unidades organizativas o roles

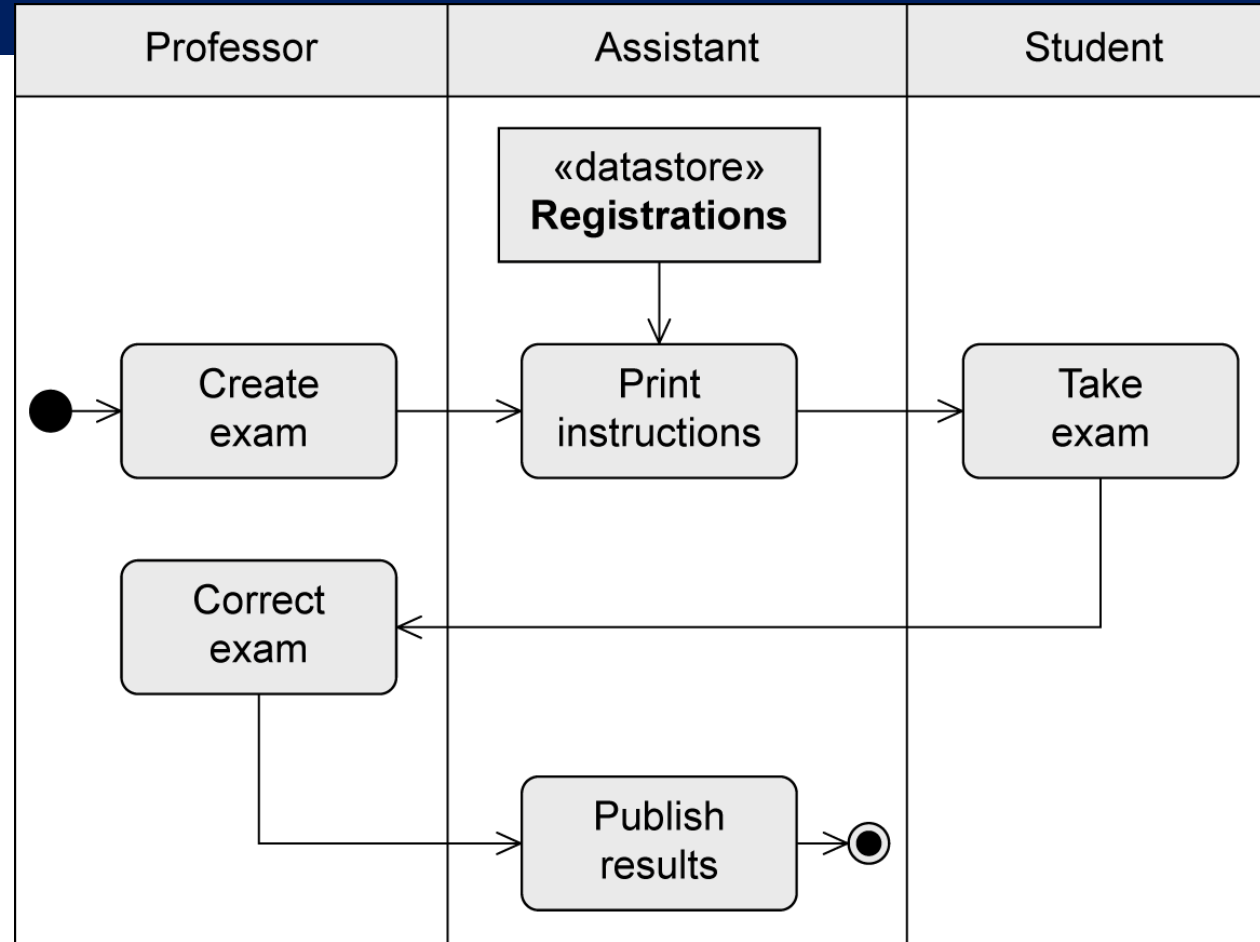
Hace que el diagrama sea más estructurado

No cambia la semántica de ejecución.

Ejemplo: particiones Estudiante y Empleado del Instituto (con subparticiones Profesor y Secretario)



# EJEMPLO



# EJEMPLO

Diagrama de máquina de estado de la identificación del estudiante:

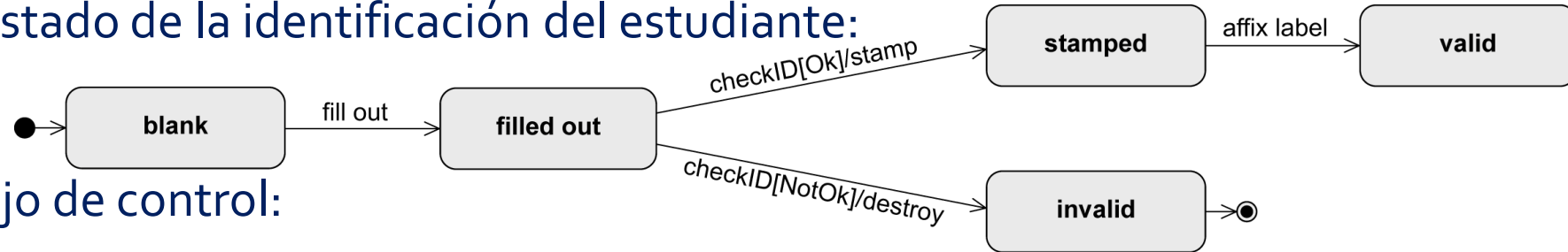
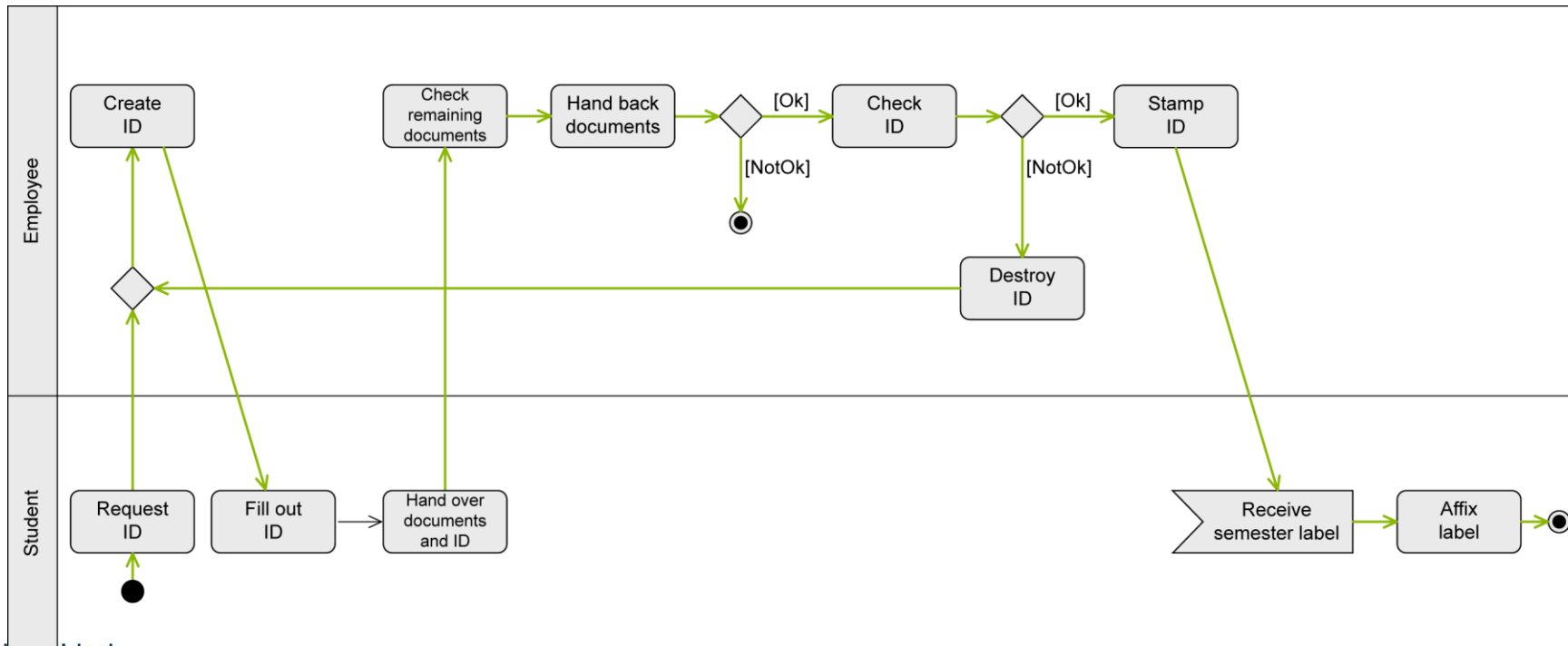


Diagrama de actividad - flujo de control:

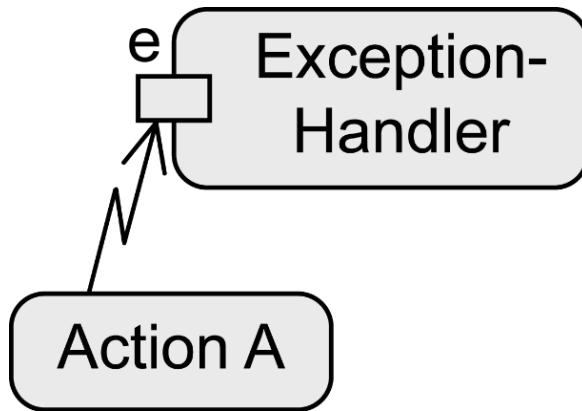


# EXCEPCIONES

Excepciones predefinidas

Definir cómo debe reaccionar el sistema en una situación de error específica.

El manejador de excepciones reemplaza la acción desde la que sucedió el error



Si se produce el error e ...

- Se eliminan todos los tokens de la Acción A
- El controlador de excepciones está activado
- El controlador de excepciones se ejecuta en lugar de la Acción A
- La ejecución luego continúa regularmente

# EJEMPLO

