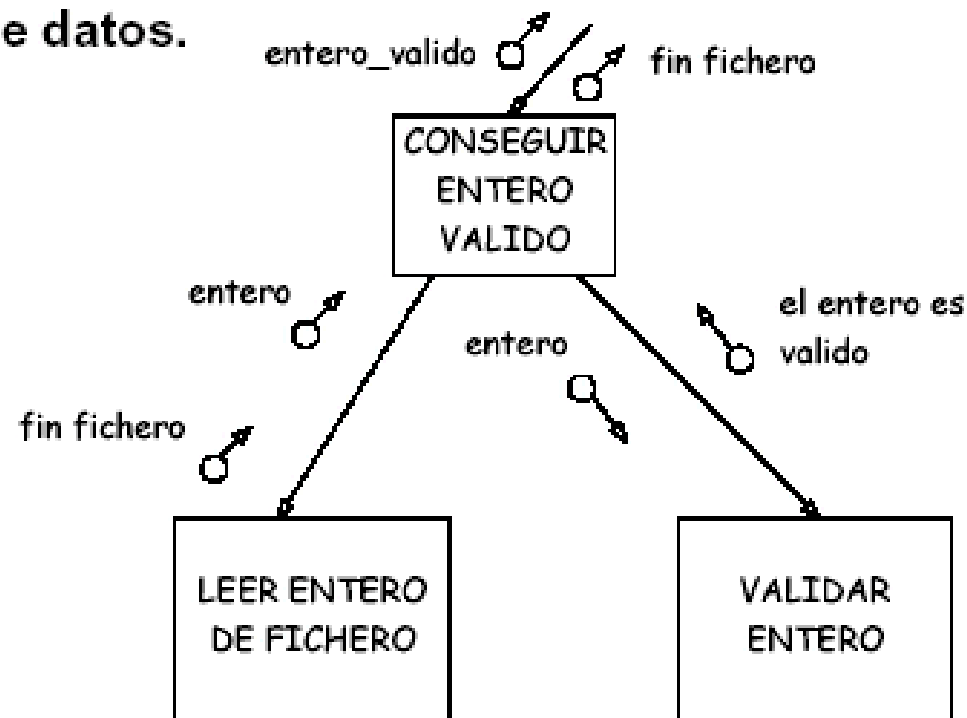


Herramientas Gráficas del Diseño Estructurado

Representación gráfica de la comunicación entre módulos

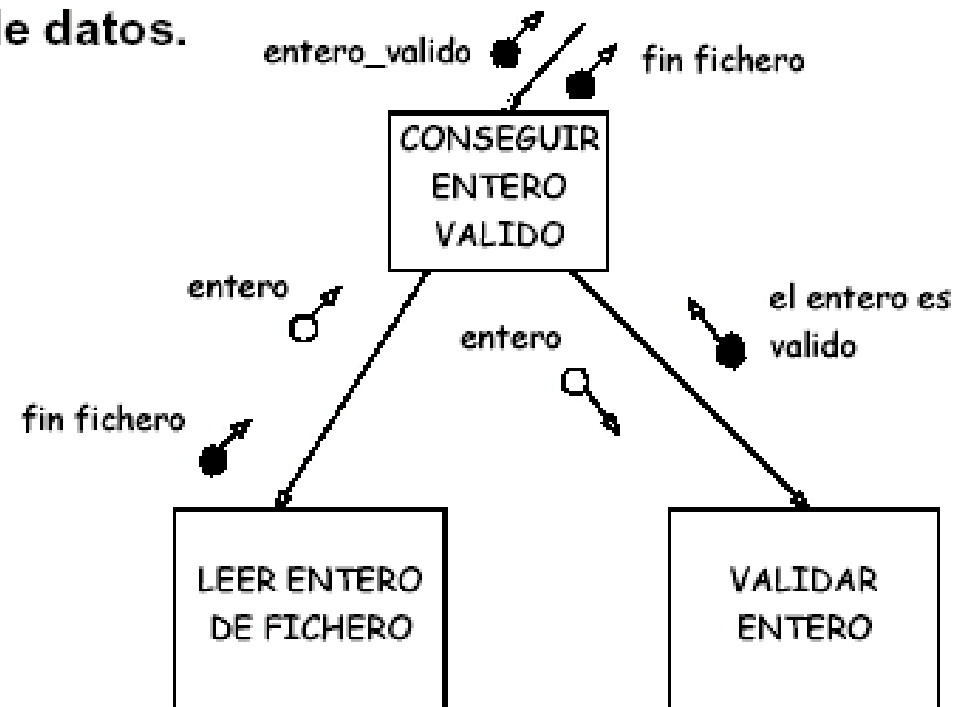
- Los datos son procesados, mientras que los flags son activados y comprobados.
- Los datos pertenecen al entorno del problema, mientras que los flags son entidades que se introducen para describir cualidades de datos.



Herramientas Gráficas del Diseño Estructurado

Representación gráfica de la comunicación entre módulos

- Los datos son procesados, mientras que los flags son activados y comprobados.
- Los datos pertenecen al entorno del problema, mientras que los flags son entidades que se introducen para describir cualidades de datos.



- **Título: Guías Adicionales de Diseño.**
- **Contenido:**
 - ◆ **Introducción.**
 - ◆ **Factorización.**
 - ◆ **Disgregación en las Decisiones.**
 - ◆ **Forma del Sistema.**
 - ◆ **Información de los errores.**
 - ◆ **Edición.**
 - ◆ **Memoria Previa.**
 - ◆ **Cúmulos de Información.**
 - ◆ **Módulos de Inicialización/Terminación.**
 - ◆ **Abanico de Entrada/Salida.**

■ **Bibliografía básica:**

- ◆ **The Practical Guide to Structured System Design. *Page-Jones*. 1988.**
- ◆ **Metodología y Tecnología de la Programación. *Sánchez P. y otros*. 1997.**
- ◆ **Ejercicios Solucionados de Metodología y Tecnología de la Programación. *Sánchez P. y otros*. 1997.**

■ **Bibliografía complementaria:**

- ◆ **Ingeniería del Software. *Pressman R*. 1993 (3ª ed.) y 1997 (4ª ed.).**
- ◆ **Software Specification and Design: A Disciplined Approach for Real-Time Systems, *Keller M. & Shumate K.*, 1992.**

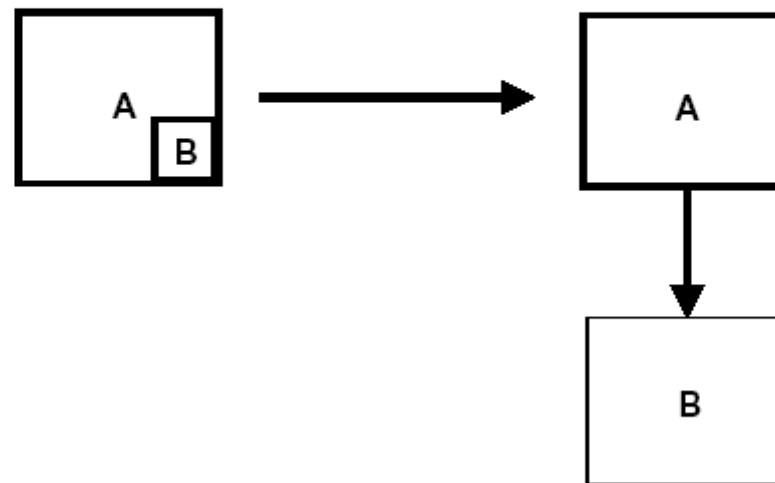
Guías Adicionales de Diseño

Introducción.

- Se debe tener en cuenta que:
 - ◆ *Cohesión y Acoplamiento* no son suficientes para el diseño de sistemas de calidad.
 - ◆ Deben considerarse otros criterios (guías) adicionales de diseño.

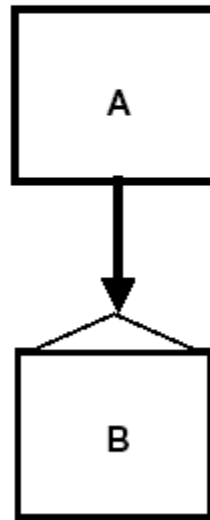
Factorización.

- *Factorizar es extraer código identificado como una función dentro de un módulo.*



Factorización.

- *Desfactorizar representa al proceso inverso.*



Guías Adicionales de Diseño

Factorización.

- *¿Por qué factorizar?:*
 - ◆ Reducción del tamaño de módulos.
 - ◆ Clarificación del Sistema.
 - ◆ Minimización del código duplicado.
 - ◆ Separación de Trabajo y Dirección.

Guías Adicionales de Diseño

Factorización: Reducción del tamaño de módulos.

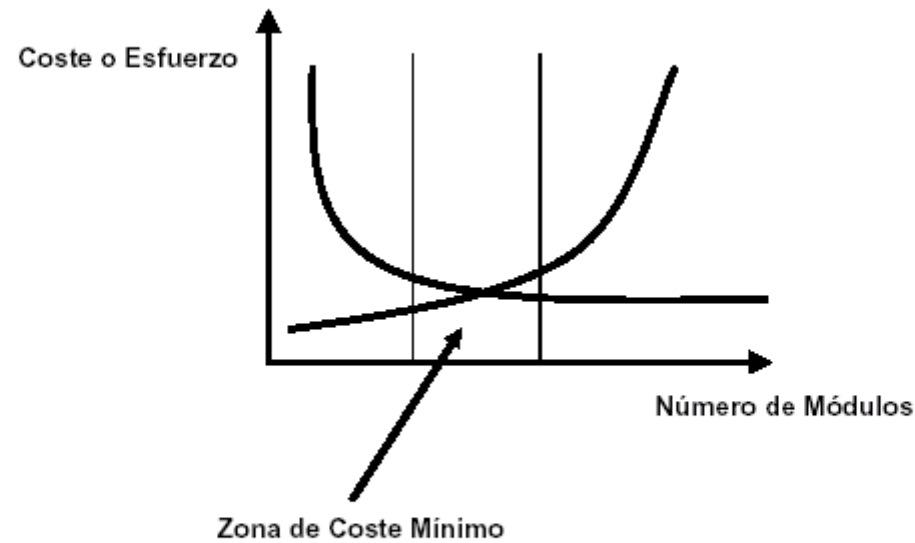
- ¿Qué tamaño debe tener idealmente un módulo?
 - ◆ *“De 60 a 120 líneas, dos hojas, de modo que se pueda ver entero de una vez”. Page Jones, 1988.*

 - ◆ *“Nuestra capacidad de entender un módulo y encontrar errores en él depende de que seamos capaces de comprender todo el módulo de una sola vez”. Jerry Weinberg, 1971.*

 - ◆ *“La claridad de un programa es inversamente proporcional al número de dedos necesarios para ser leído”. Tom DeMarco, 1978.*

Guías Adicionales de Diseño

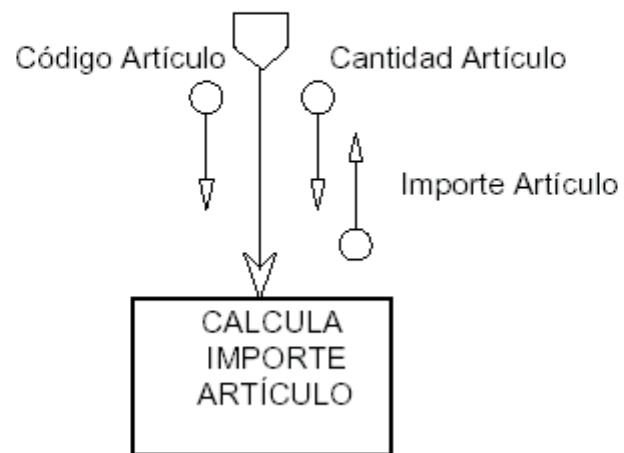
Factorización: Reducción del tamaño de módulos.



- Se deja de factorizar:
1. Cuando no se encuentre una función bien definida
 2. Las interfaces sean tan complejas como los propios módulos

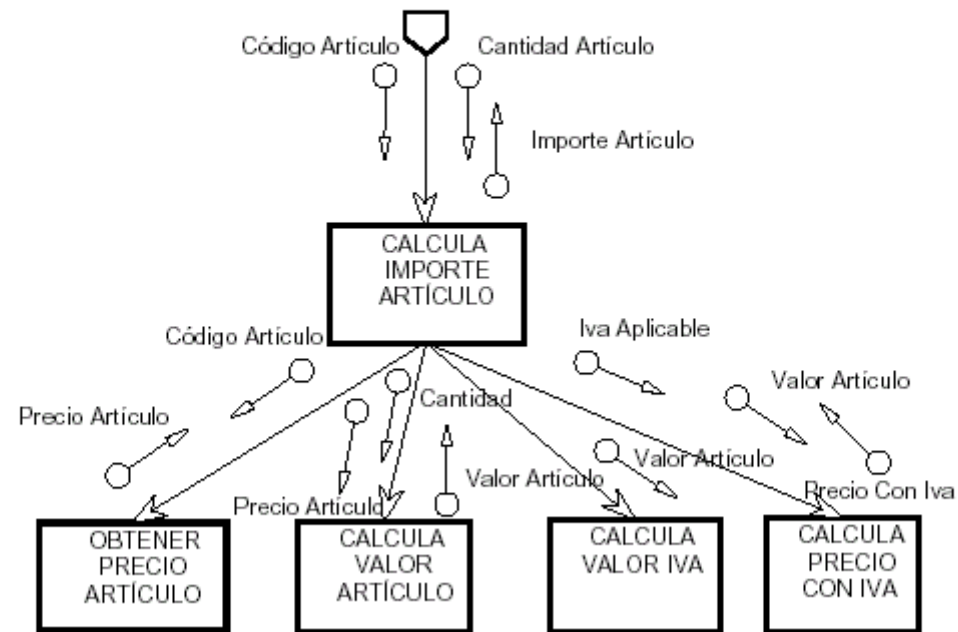
Guías Adicionales de Diseño

Factorización: Clarificación del Sistema.



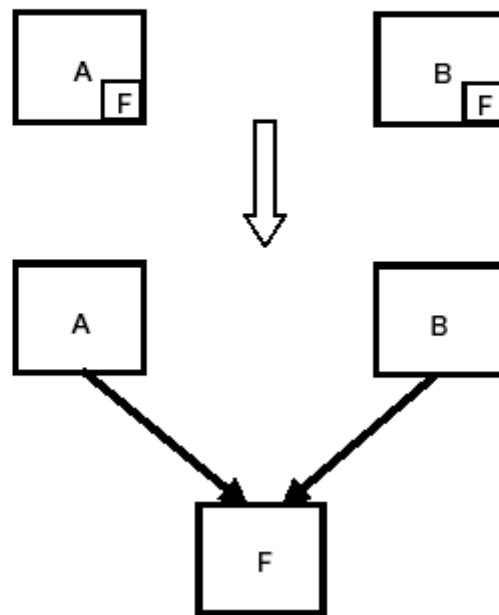
Guías Adicionales de Diseño

Factorización: Clarificación del Sistema.



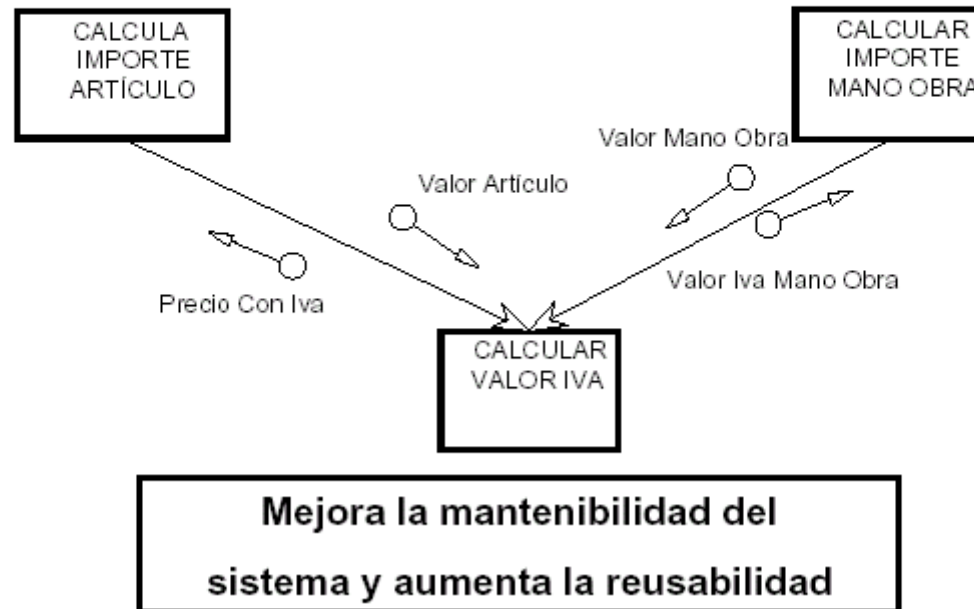
Guías Adicionales de Diseño

Factorización: Minimización del Código duplicado.



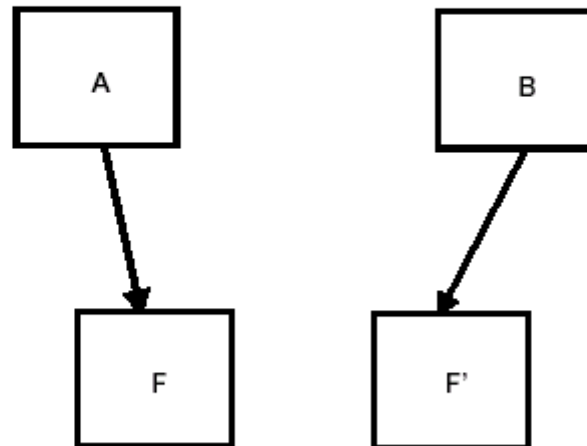
Guías Adicionales de Diseño

Factorización: Minimización del Código duplicado.



Guías Adicionales de Diseño

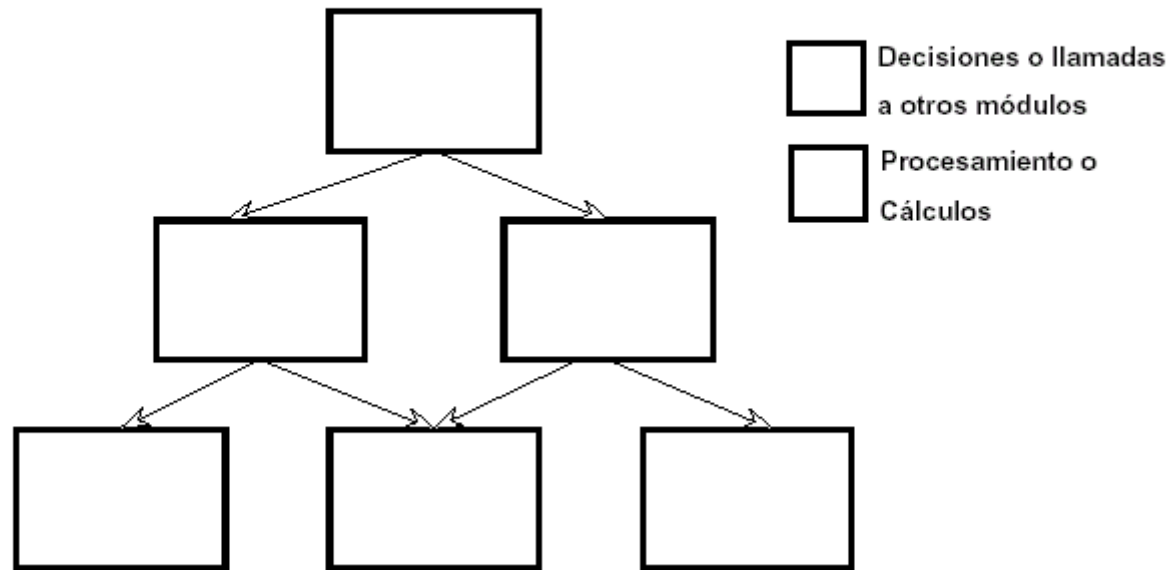
Factorización: Minimización del Código Duplicado.



Si la funcionalidad es distinta crear entonces módulos separados

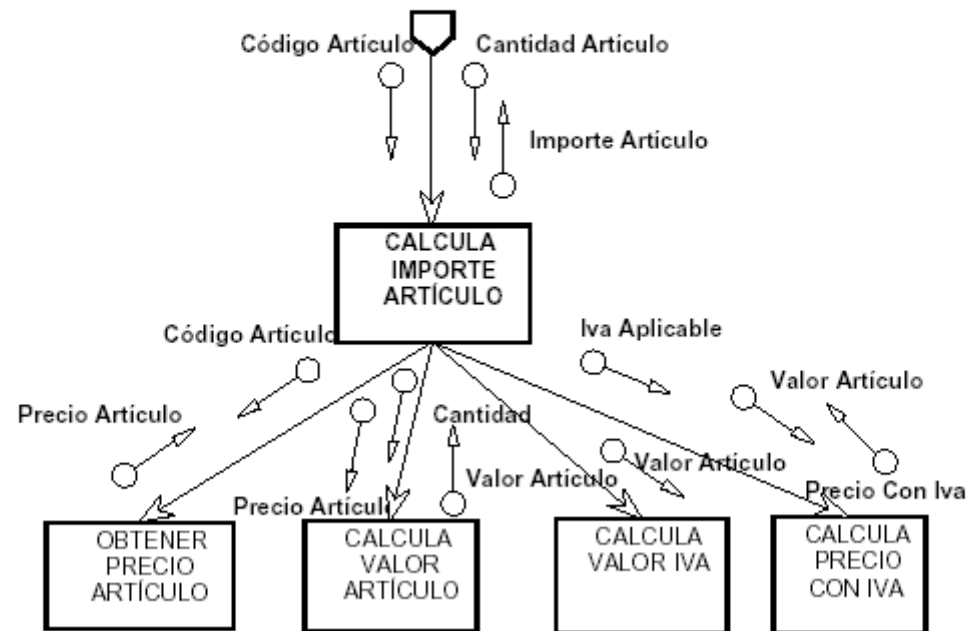
Guías Adicionales de Diseño

Factorización: Separación de Trabajo y Dirección.



Guías Adicionales de Diseño

Factorización: Separación de Trabajo y Dirección.



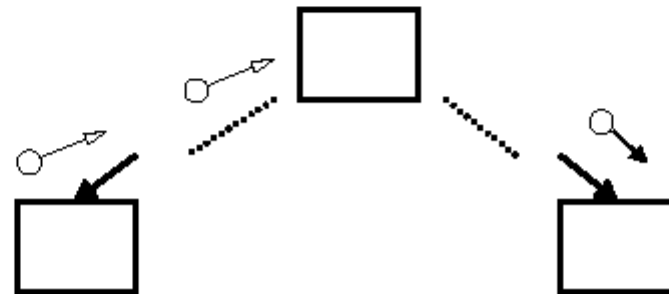
Guías Adicionales de Diseño

- ◆ **Introducción.**
- ◆ **Factorización.**
- ◆ **Disgregación en las Decisiones.**
- ◆ **Forma del Sistema.**
- ◆ **Información de los errores.**
- ◆ **Edición.**
- ◆ **Memoria Previa.**
- ◆ **Cúmulos de Información.**
- ◆ **Módulos de Inicialización/Terminación.**
- ◆ **Abanico de Entrada/Salida.**
- ◆ **Realización de un ejercicio.**

Guías Adicionales de Diseño

Disgregación de Decisiones.

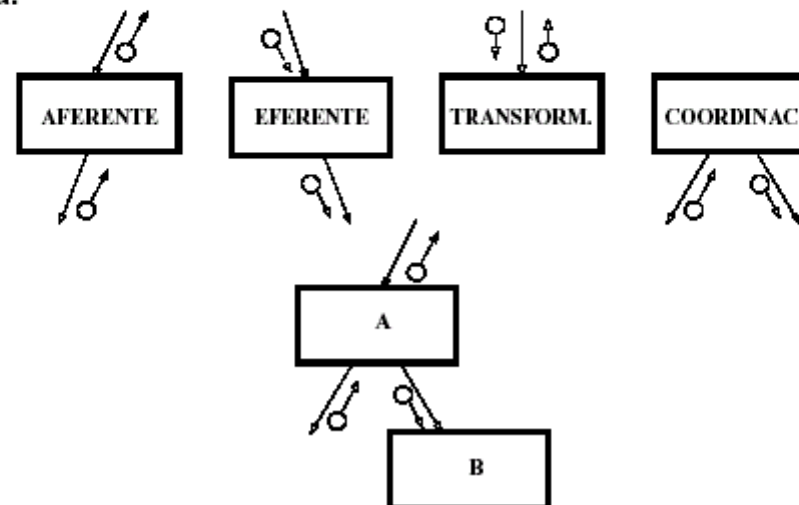
- Una decisión tiene dos partes:
 - ◆ La parte que se ocupa del reconocimiento de la acción que se va a ejecutar.
 - ◆ La parte de ejecución de la acción.
- Se da Disgregación en la Decisión cuando la parte de reconocimiento y la parte de ejecución no están en el mismo módulo.
- Síntomas: Flujos de Datos Vabagundos.



Guías Adicionales de Diseño

Forma del Sistema.

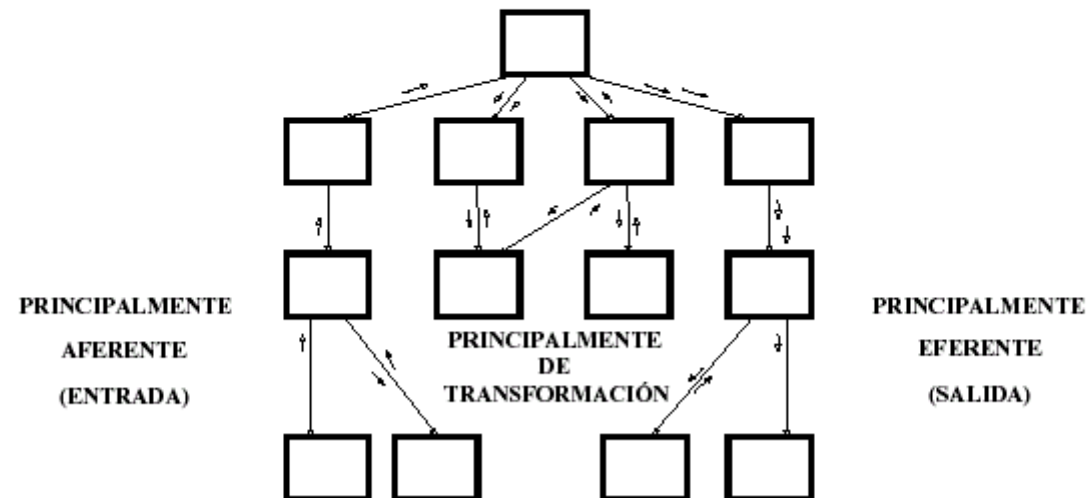
- La estructura general del Diagrama de Estructura dice bastante sobre la *calidad* del sistema diseñado.
- Se distinguen 4 tipos de módulos que determinan la *forma* de un sistema:



Guías Adicionales de Diseño

Forma del Sistema.

- Se distinguen tres partes diferenciadas en un Diagrama de Estructura: *rama principalmente aferente*, *rama principalmente de transformación* y *rama principalmente eferente*.



Guías Adicionales de Diseño

Forma del Sistema: Sistemas Dirigidos por la Entrada Física.

- **Sistemas Dirigidos por la Entrada Física:** Aquellos con poco procesamiento en su parte aferente de forma que los módulos superiores del DE deben tratar información no depurada.

- **Problemas típicos:**
 - ◆ Acoplamiento alto debido a la disgregación en la decisión.
 - ◆ Cambios en la estructura física de los datos implican cambios en los módulos superiores del DE.
 - ◆ Reusabilidad casi nula.

- El problema puede presentarse en la salida: *Sistemas dirigidos por la salida física.*

Guías Adicionales de Diseño

Forma del Sistema: Sistemas Balanceados.

- Un *Sistema Balanceado* es tal que en los módulos superiores se gestiona información de naturaleza más lógica que física.
- Un sistema balanceado no está dirigido por la entrada ni por la salida física.
- Los sistemas balanceados son:
 - ◆ Más fáciles de implementar.
 - ◆ De menos acoplamiento.
 - ◆ Más fáciles de mantener.
 - ◆ Más reusables.

Guías Adicionales de Diseño

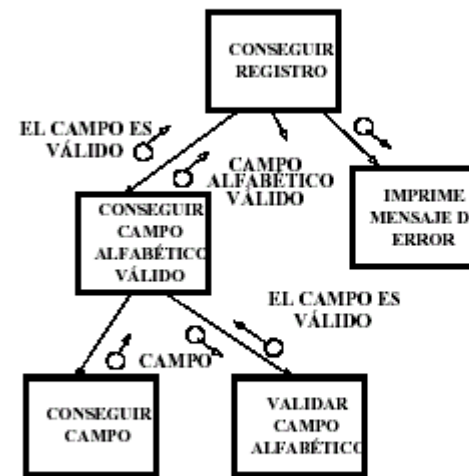
Información de los errores.

- Dos cuestiones importantes:
 - ◆ ¿Qué módulo llama al módulo que escribe los mensajes de error?
 - ◆ ¿Dónde se coloca el texto de los mensajes de error?

- Los errores deben ser informados por los módulos que *detecten* y que *conozcan* el error.

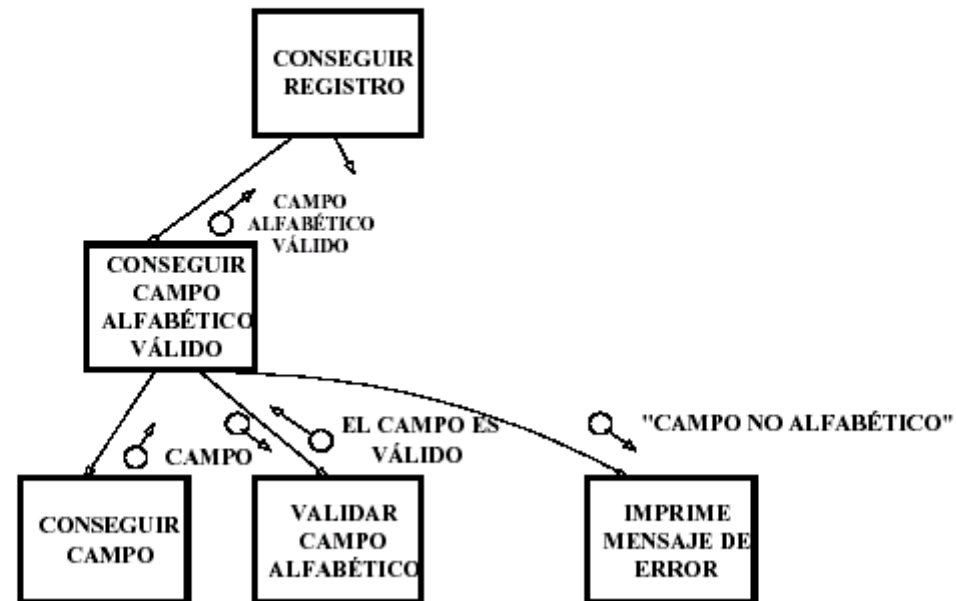
Guías Adicionales de Diseño

Información de los errores.



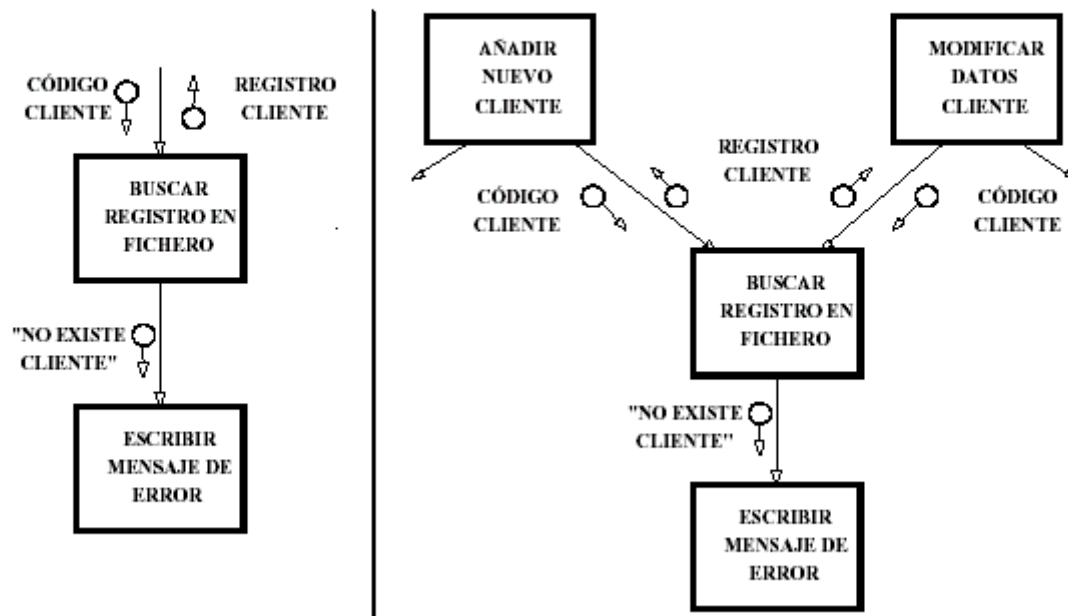
Guías Adicionales de Diseño

Información de los errores.



Guías Adicionales de Diseño

Información de los errores.



Guías Adicionales de Diseño

Información de los errores.



Guías Adicionales de Diseño

Información de los errores.

- ¿Dónde se coloca el texto de los mensajes de error?:
 - ◆ Diseminados por el código.
 - ◆ En un mismo módulo.
- Ventajas de ubicarlo en un mismo módulo.
 - ◆ Formato de mensajes consistente.
 - ◆ Almacenables fuera del código fuente.
 - ◆ Se evitan mensajes duplicados.
 - ◆ El mantenimiento es más simple.
- Pero las desventajas existentes son:
 - ◆ Aparece un identificador artificial del error.
 - ◆ El código fuente es menos legible.

Guías Adicionales de Diseño

Edición.

- La edición, o captura de datos validados, debe hacerse de acuerdo con las siguientes guías:
 - ◆ Validar la sintaxis antes que la semántica.
 - ◆ Validar lo simple antes que lo cruzado.
 - ◆ Validar lo interno antes que lo externo.