

Escuela Técnica Superior de Ingeniería Informática
Universitat Politècnica de València
Grado en Ingeniería Informática, Curso 2013-2014

Calidad de Software

Tema 3: Métricas del Software

Departamento de Sistemas Informáticos y Computación
Universitat Politècnica de València, España
email: sabrahao@dsic.upv.es



Objetivos

- Presentar de forma intuitiva el concepto de métricas del software
- Presentar una ontología de medición que formaliza el conocimiento sobre la medición del software.
- Presentar el framework QGM (Goal-Question-Metrics)
- Presentar los principales métodos de validación teórica de métricas
- Presentar los principales métodos de validación empírica de métricas

Contenido

- Tema 1: Introducción
 - Fundamentos de la Calidad del Software
- Tema 2: Perspectivas de Calidad
 - 2.1 Calidad de Producto**
 - Estándares de Calidad de Producto
 - Modelos de Calidad
 - Calidad de Arquitecturas Software
 - 2.2 Calidad de Datos**
 - Características de calidad de datos
 - 2.3 Calidad de Proceso**
 - Modelos y Estándares de Evaluación y Mejora de Proceso
 - 2.4 Calidad en Uso**
 - Estándares de Calidad en Uso
 - Métodos de Evaluación de Usabilidad

Contenido

■ Tema 3: Métricas del Software

- El paradigma Objetivo/Pregunta/Métrica (GQM)
- Definición de Métricas de Calidad
- Validación Teórica
- Validación Empírica

■ Tema 4: Gestión, Aseguramiento y Control de la Calidad

- Gestión de la Calidad del Software
- Aseguramiento de la Calidad del Software: actividades y métodos
- Control de la Calidad

Contenido

- Tema 5: Estimación de Proyectos de Software
 - Conceptos
 - Problemática de la estimación de proyectos
 - Métodos de Medición de Tamaño Funcional
 - FPA, COSMIC
 - Generación de indicadores de productividad, calidad, etc.
 - COCOMO II

Métricas del Software

- Introducción
- Goal-Question-Metric (GQM)
- Validación Teórica
- Validación Empírica



Métricas del Software

- Las métricas son un buen medio para entender, monitorizar, controlar, predecir y probar el desarrollo de software y los proyectos de mantenimiento (Briand et al., 1996).
- En general, la **medición** persigue tres objetivos fundamentales (Fenton y Pfleeger, 1997):
 - entender qué ocurre durante el desarrollo y el mantenimiento
 - controlar qué es lo que ocurre en los proyectos de desarrollo
 - mejorar los procesos y productos
- Las métricas pueden ser utilizadas para que los profesionales puedan tomar las mejores decisiones.



**MEDIDAS COMO MEDIOS PARA ASEGURAR LA CALIDAD EN
LOS PRODUCTOS SOFTWARE**

Métricas del Software

- Sin embargo, la **medición** ha sido ignorada en la Ingeniería del Software:
 - Todavía fallamos en dar **objetivos medibles** cuando desarrollamos productos software.
 - Por ejemplo, se dice que será amigable, fiable y mantenible, sin especificar qué significa esto en términos medibles.
 - Fallamos al medir diferentes componentes que permiten calcular los costes reales de los proyectos software.
 - Por ejemplo, normalmente no sabemos cuánto tiempo fue realmente invertido en el diseño, comparado con las pruebas.
 - No intentamos cuantificar la calidad de los productos que producimos.
 - Por ejemplo, no podemos decir a un usuario cómo de fiable va a ser un producto en términos de fallos en un periodo dado de uso.

Métricas del Software

- Solemos ver informes que hacen afirmaciones como que el **80% de los costes del software son de mantenimiento** o que hay una **media de 55 errores en cada 1.000 líneas de código**.
- Sin embargo, no se dice:
 - Cómo se obtuvieron esos resultados,
 - Cómo se diseñaron y ejecutaron los experimentos,
 - Qué entidades fueron medidas y cómo y
 - Cuales fueron los márgenes de error.
- Sin estos datos no podemos **repetir las mediciones de forma objetiva** en otros entornos para tener comparaciones con los estándares de la industria.

Métricas del Software

- Todos estos problemas derivados de una medición insuficiente se agravan por una falta de una aproximación rigurosa a la medición.
- En general, la producción software está en crisis, tiene costes excesivos, baja productividad y poca calidad.
- Se ha llegado a sugerir que esto es debido a que no medimos.
- En software hay **tres clases de entidades** cuyos **atributos** podemos medir:
 - **Procesos**: actividades del desarrollo del software que normalmente conllevan el factor tiempo.
 - **Productos**: entregables, artefactos o documentos generados en el ciclo de vida del software.
 - **Recursos**: todos aquellos elementos que hacen de entrada a la producción del software.

Atributos a medir

■ Procesos:

- El tiempo (duración del proceso)
- El esfuerzo (asociado al proceso)
- El número de incidentes de un tipo específico que se dan durante el proceso (Ej., el número de errores de requisitos encontrados durante la construcción de la especificación)

■ Productos:

- La fiabilidad del código
- La entendibilidad de un documento de especificación
- La mantenibilidad del código fuente
- La longitud, funcionalidad, modularidad o corrección sintáctica de los documentos de especificación

■ Recursos:

- El personal
- Los materiales
- Las herramientas y métodos
- El coste
- La productividad

Ámbito/Utilidad de las Métricas del Software

- **Estimación de Esfuerzo y Costes:** proceso de predecir la cantidad de esfuerzo requerida para construir un producto software. Ejemplo: COCOMO, SLIM. etc.
- **Modelos y medidas de productividad:** el ratio de salida por unidad de entrada. *Productividad = tamaño / esfuerzo* o *Productividad = LOC / personas-mes*.
- **Modelos y medidas de calidad:** medición de calidad de software. Modelo de McCall, ISO/IEC 9126, SQUARE, etc..
- **Modelos de fiabilidad:** Plot de cambios de intensidad de fallos por tiempo. Ejemplo: Modelo exponencial básico, Modelo logaritmo Poisson.
- **Modelos y evaluación de rendimiento:** uso de características de rendimiento observables (tiempo de respuesta y tasa de complejión).
- **Métricas de complejidad estructural:** complejidad ciclomática,
- ...



Ejemplo

- Una empresa quiere iniciar un nuevo servicio de descargas de vídeos y juegos bajo demanda. El servicio será proporcionado a usuarios que tengan PCs y se registran al servicio.
- Los clientes deben usar software especializado para bajar vídeos y juegos del servidor. La **intensidad de fallos** del software es de **1 fallo por 100 CPU/hr**. El sistema software especializado ejecuta **20 CPU/hr por semana en cada máquina cliente** y hay **800 clientes** que usarán el servicio.
- Se quiere proporcionar a los clientes un servicio de reparación. Cada empleado puede realizar 4 llamadas de servicio por día y el servicio está disponible 5 días por semana.
- ¿Cuántos empleados necesitamos contratar?

Ejemplo

- ¿Cuántos empleados necesitamos contratar?
- Usando el valor de la intensidad de fallos, cada sistema experimenta **0,2 fallos por 20 horas de operación** (o 0,2 fallos por semana de media).
- El total de fallos para 800 clientes es de 160 por semana o 32 por día.
- Cada empleado puede realizar 4 visitas por día. Así, el número de personal es de **$32/4 = 8$** .

¿Qué es la Medición del Software?

- **Medición** es el proceso por el cuál se asignan números o símbolos (**medidas**) a los **atributos** de las **entidades (objetos)** del mundo real.

$$\text{Object} = \left\{ \begin{array}{ll} \text{attribute}_1 & (\text{value}_{11}, \text{value}_{12}, \dots) \\ \text{attribute}_2 & (\text{value}_{21}, \text{value}_{22}, \dots) \\ \dots & \dots \\ \text{attribute}_n & (\text{value}_{n1}, \text{value}_{n2}, \dots) \end{array} \right\}$$

¿Qué es la Medición del Software?

- **Medición del Software:**

“Una función que toma como entrada cierta información del software que se está midiendo, y que devuelve como salida un valor numérico, el cual es interpretado como el grado en que el producto software posee un atributo dado que afecta a su calidad”. (IEEE:1992)

- **Métrica:**

“Un **método** de medición y una **escala cuantitativos** que pueden ser usados para determinar el valor que toma cierta característica en un producto software concreto”.

(ISO/IEC 14598-1:1999)

¿Qué es la Medición del Software?

- La medición de software es una disciplina relativamente joven, y no existe consenso general sobre la definición exacta de los conceptos y terminología que maneja.
- Creación de una **ontología de medición** entre:
 - Universidad de Castilla-La Mancha
 - Universidad de Málaga
 - Universidad Nacional de La Pampa
 - Universidad Politécnica de Cataluña
 - Universidad Politécnica de Valencia

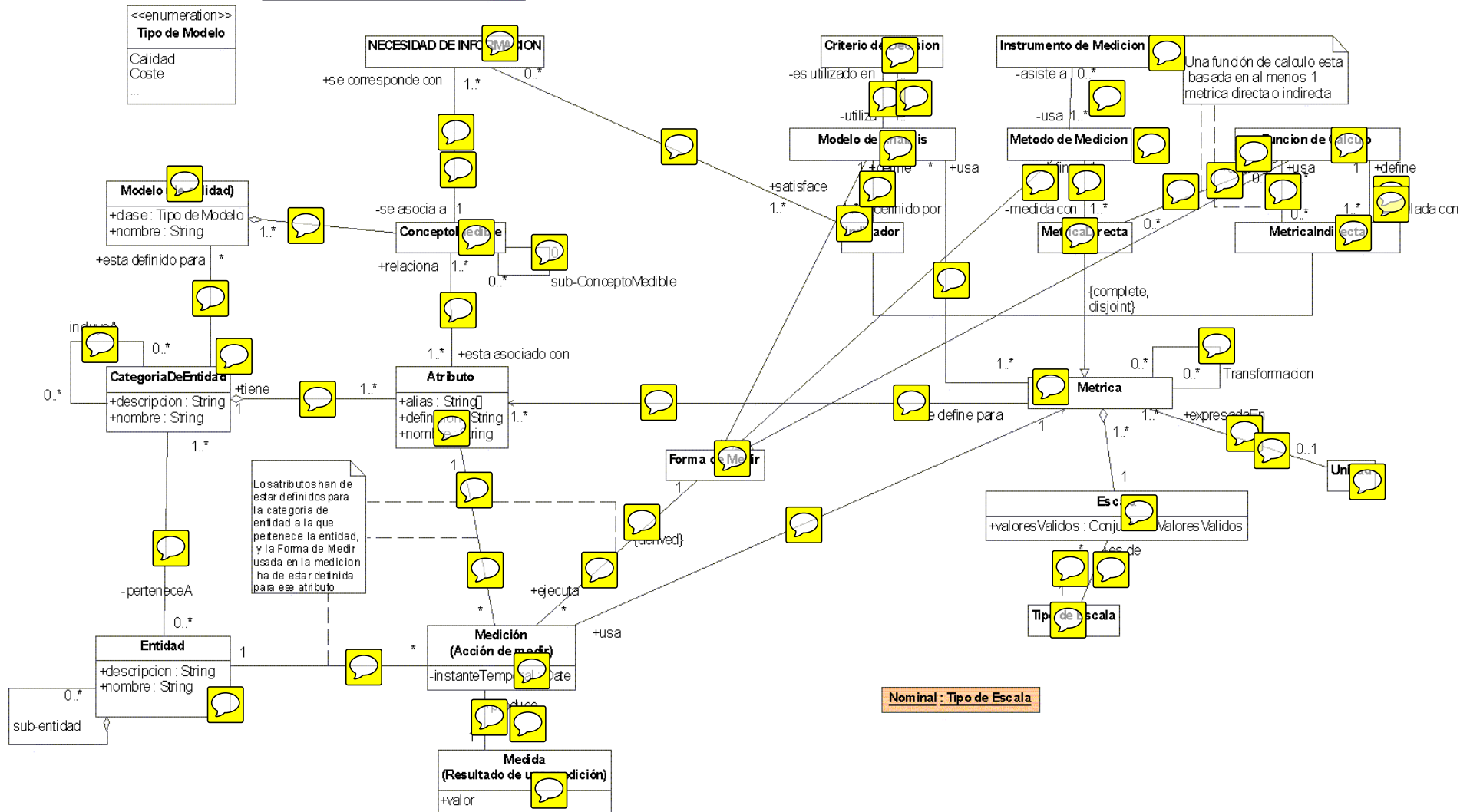


Ontología de la Medición

- Qué es una ontología? (Gruber, 1995)
 - Una ontología es una especificación de una conceptualización.
- En otras palabras:
 - Mediante la definición de ontologías se pretende reunir y formalizar el conocimiento sobre un determinado dominio de problema.
 - Se mejora el entendimiento y la comunicación mediante el establecimiento de vocabularios comunes, lo que facilita la reutilización y la interoperabilidad de los sistemas software.

Ontología de la Medición

ONTOLÓGIA DE MEDICIÓN DEL SOFTWARE V5 (28/11/03)



Ontología de la Medición

- **Concepto. ATRIBUTO**
- **Definición.** Una propiedad mensurable, física o abstracta, que comparten todas las entidades de una categoría de entidad.
- **Relaciones.**
 - Una medición se realiza sobre los atributos de una entidad
 - Un atributo tiene definida cero, una o varias *métricas*.
 - Un atributo sólo puede pertenecer a una *categoría de entidad*.
 - Un atributo está relacionado con uno o más conceptos medibles.
- **Ejemplos**
 - El *atributo* “**tamaño de código fuente**”, de “**programas en C**” que es diferente del *atributo* de “**programa en Ada**”.

Ontología de la Medición

- **Concepto:** MÉTRICA
- **Definición.** Una forma de medir y una escala, definidas para realizar mediciones de uno o varios atributos
- **Relaciones:**
 - Una métrica está definida para uno o más atributos
 - Dos *métricas* pueden relacionarse mediante una función de transformación.
 - El tipo de dicha función de transformación va a depender del tipo de escala de ambas métricas.
 - Una métrica puede expresarse en una unidad (sólo para métricas cuya escala sea de tipo intervalo o ratio)
- **Ejemplos**
 - “**líneas de código**” para el “**tamaño**” de un “**módulo en C**” o de un “**programa en Ada**”.

Ontología de la Medición

- **Concepto.** MEDIDA
- **Definición.** Resultado de una *medición*.
- **Relaciones**
 - Una medida es el resultado de una medición
- **Ejemplos**
 - 35.000 líneas de código, 200 páginas, 50 clases.
 - 5 meses desde el comienzo al fin del proyecto.
 - 0,5 fallos por cada 1.000 líneas de código.

Ontología de la Medición

■ Ejemplo 1:

- Atributo: tamaño de código fuente
- Entidad: programa C
- Métrica: Líneas de Código (LCF)
- Medida: 10.000 LCF

■ Ejemplo 2:

- Atributo: tamaño
- Entidad: diagrama de clases UML
- Métricas: Número de clases.
- Medida: 30 clases

■ Métricas Directas: una métrica de un atributo que no depende de ninguna métrica de otro atributo

- LCF (líneas de código fuente escritas)
- HPD (horas-programador diarias)
- CHP (coste por hora-programador, unid. monetarias)
- #def (numero de defectos en la fase de pruebas)

■ Métricas Indirectas: una métrica de un atributo que se deriva de una o más métricas de otros atributos.

- LCFH: líneas de código fuente por hora de programador
- CLCF (coste por línea de código fuente) = LCF / CTP
- EffDef (Eficiencia detección defectos) = $\#def \text{ detectados} / \#total \text{ defectos}$

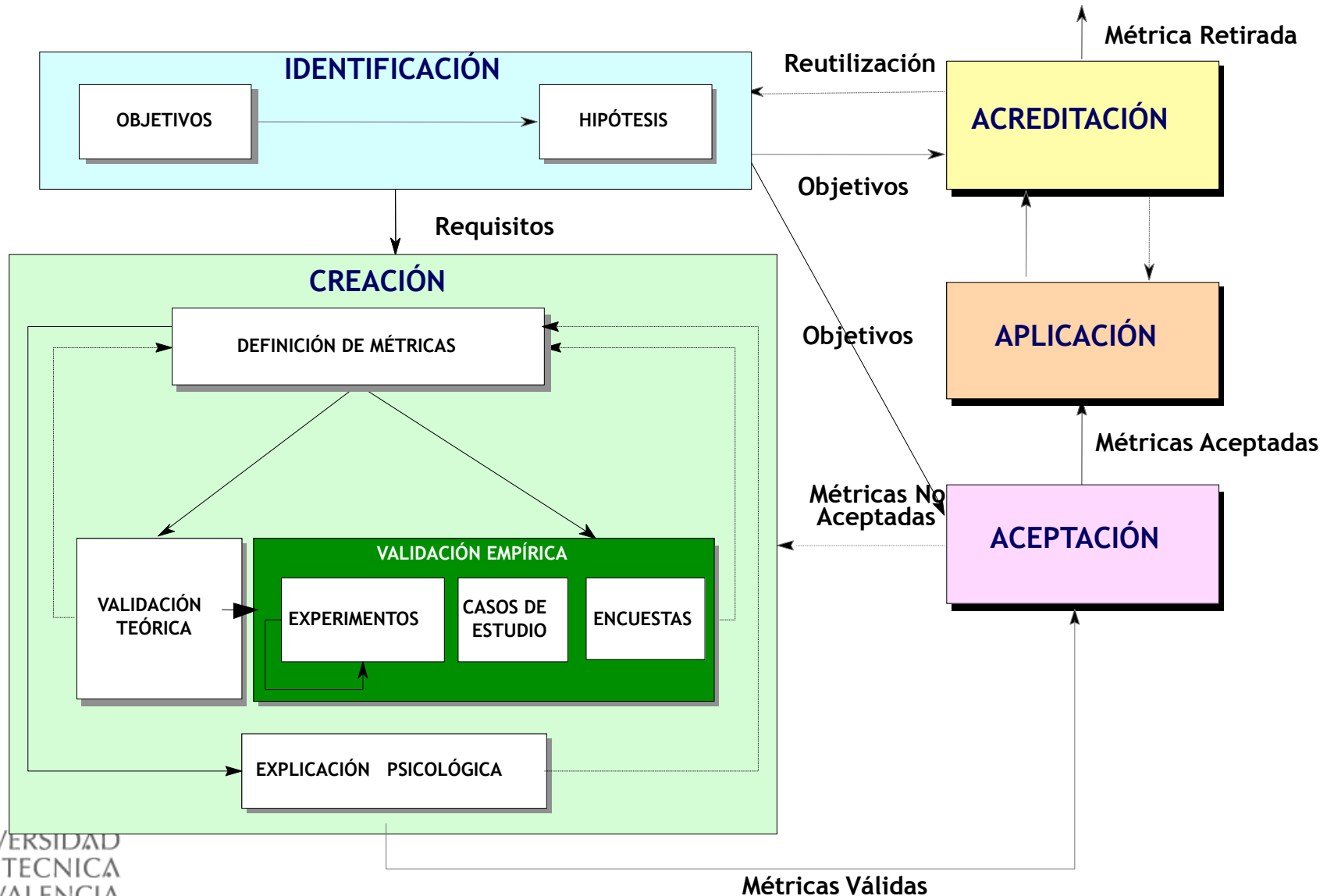
Métricas

- Existe un gran número de métricas aunque muy pocas van más allá de su definición y no se usan en la industria.
- Esto se debe a múltiples problemas, entre ellos:
 - Las métricas no se definen siempre en el contexto del objetivo de interés industrial que se pretende alcanzar.
 - No siempre es posible realizar una **validación teórica** adecuada de la métrica porque el atributo que queremos medir no siempre está bien definido.
 - Un gran número de métricas nunca se ha **validado empíricamente**

Métricas

- Esta situación ha conducido ambigüedad en:
 - las definiciones,
 - propiedades y
 - asunciones de las métricas,
- haciendo que:
 - el uso de las mismas sea difícil,
 - la interpretación peligrosa y
 - los resultados contradictorios.
- Para evitarlo es necesario contar con un **método de definición de métricas**.

Método de Definición de Métricas



Método de Definición

Identificación

- Se **definen los objetivos** que se persiguen a la hora de crear la métrica.
- Se **plantean las hipótesis** de cómo se llevará a cabo la medición.
- Sobre los elementos de esta etapa (objetivos e hipótesis) se basarán todas las etapas siguientes.
- Como resultado de esta etapa se generan los requisitos que debe cumplir la métrica.

Método de Definición

Creación

- Hay que tener en cuenta las características del sistema junto con la experiencia de los diseñadores.
- Hay que tener un **objetivo concreto** para evitar la definición de métricas que en realidad no capturen el objetivo perseguido (GQM: *Goal-Question-Metric**).

* Basili and Rombach's Goal-Question-Metrics paradigm, *IEEE Transactions on Software Engineering*, 1988 paper on the TAME project.

Goal-Question Metric (GQM)

- Introducción
- Ejemplo

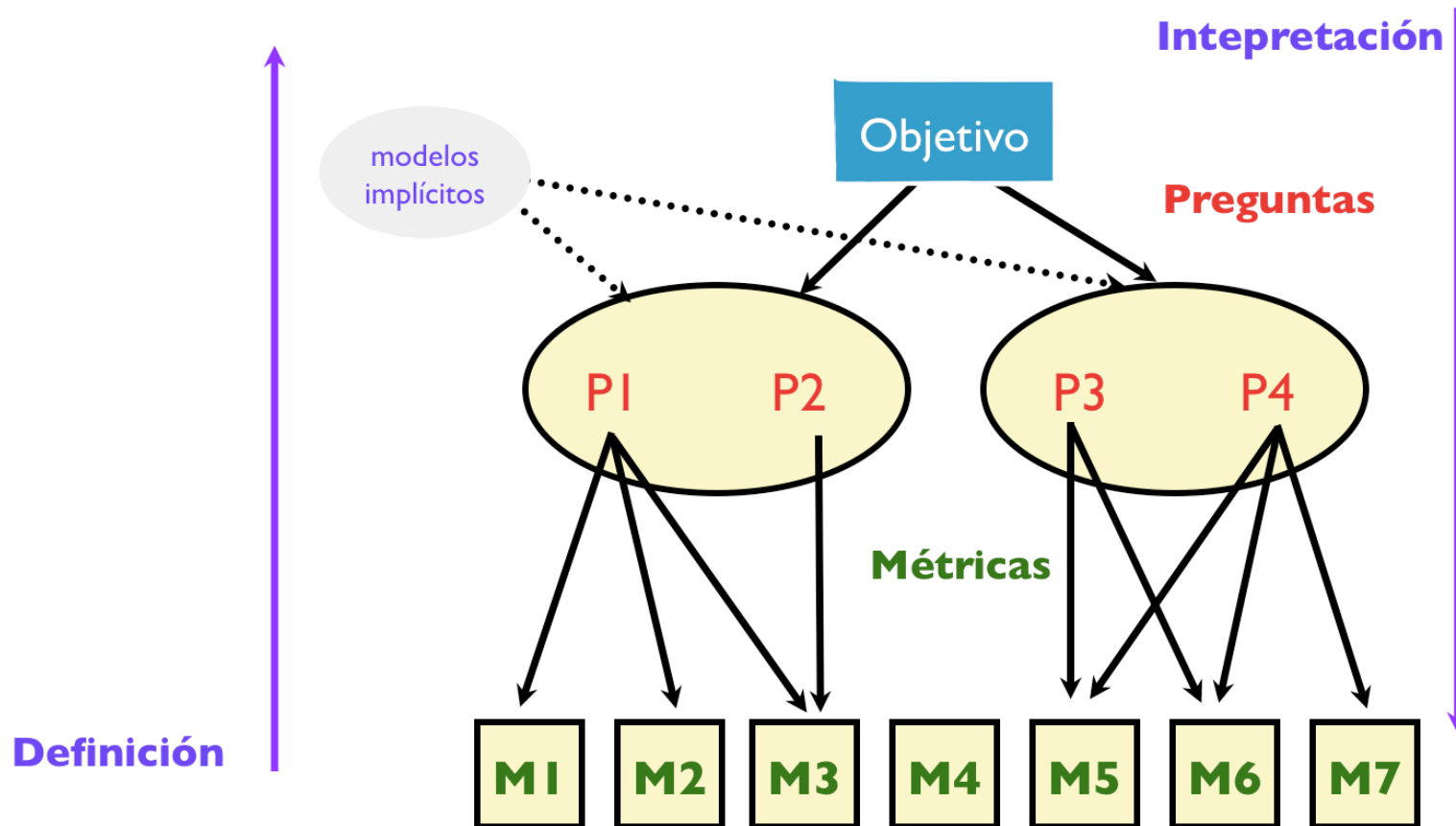


UNIVERSIDAD
POLITECNICA
DE VALENCIA

Goal Question Metric (GQM)

- Es una técnica definida por Basili y Weiss (1984) y Rombach (1990), para seleccionar y generar métricas tanto del proceso como de los resultados de un proyecto.
- Principio básico: la medición debe ser realizada siempre orientada a un objetivo.
- Define un **objetivo**, el cual se refina en **preguntas** y define **métricas** que intentan dar información para responder a estas preguntas.
- Conformado por 3 niveles:
 - Nivel conceptual (Objetivos - *Goals*)
 - Nivel operacional (Preguntas - *Questions*)
 - Nivel cuantitativo (Métricas - *Metrics*)

Goal Question Metric (GQM)



Goal-Question-Metric

■ Definir los objetivos de la medición

Analizar	el objeto bajo medición
Con el propósito de	entender, controlar, o mejorar el objeto
Con respecto a	el enfoque de calidad del objeto en el que se centra la medición
Desde el punto de vista de	las personas que miden el objeto
En el contexto de	el entorno en el que la medición tiene lugar

■ Definir preguntas e hipótesis

- ☐ Con la respuesta a las preguntas planteadas, se debería poder concluir si se cumple un determinado objetivo.

■ Definir las métricas

- ☐ Deben proporcionar información cuantitativa que permita responder las preguntas planteadas.

Ejemplo 1

Métricas para BD Relacionales

■ Objetivo GQM:

- ☐ *Analizar* BD Relacionales
- ☐ *Con el propósito de* Asegurar
- ☐ *Con respecto a la* Mantenibilidad
- ☐ *Desde el punto de vista* De los diseñadores de BD
- ☐ *En el contexto de* BD Relacional

■ Preguntas:

- ☐ Pregunta 1: ¿Cómo influye la complejidad de las tablas en la mantenibilidad de las Bases de Datos relacionales?
- ☐ Pregunta 2: ¿Cómo influye la complejidad entre tablas en la mantenibilidad de las Bases de Datos relacionales?

Ejemplo 1

Métricas para BD Relacionales

■ Métricas:

■ **Pregunta 1**

- **NA(T)** - NÚMERO DE ATRIBUTOS DE UNA TABLA
- **NFK(T)** - NÚMERO DE CLAVES AJENAS
- **RFK(T)** - RATIO DE CLAVES AJENAS DE UNA TABLA

$$RFK(T) = \frac{NFK(T)}{NA(T)}$$

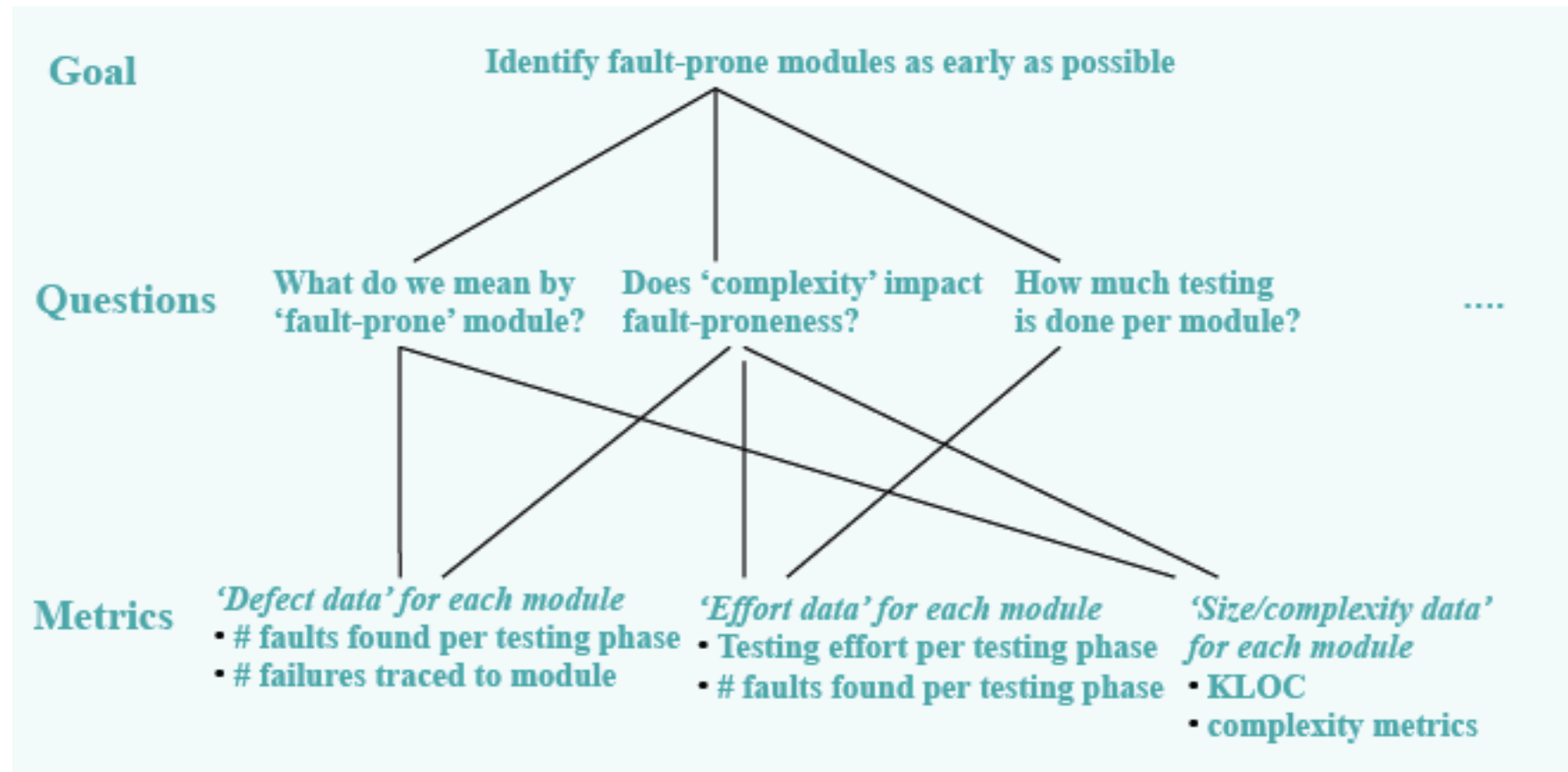
■ **Pregunta 2**

- **NT** - NÚMERO DE TABLAS
- **NA** - NÚMERO DE ATRIBUTOS
- **NFK** - NÚMERO DE CLAVES AJENAS (NFK)



Ejemplo 2

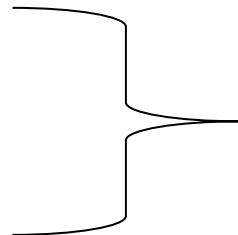
Identificación de módulos propensos a errores



Las Escalas de Medida

Cuatro grandes tipos:

- Nominal
- Ordinal
- Intervalo
- Razón



- El tipo de escala de medida que adoptan los datos, determina el **tipo de operaciones aritméticas** que se pueden realizar con ellos y, por tanto, también el **tipo de análisis estadístico**.
- Las escalas de intervalo y de razón aparecen, a veces, definidas como “cuantitativas”.



Escala Nominal

- Sólo permiten la **clasificación** o **diferenciación** de los objetos.
- Permiten establecer relaciones de igualdad o desigualdad entre dos o más objetos.
- Las variables que adoptan este nivel de medida se denominan “cualitativas”.
- Con los “números” de este tipo de variables **no** se pueden efectuar operaciones aritméticas.
- Ejemplo:
 - Clasificación de fallos en el software (fallo de especificación, fallo de diseño, fallo de codificación)

Escala Ordinal

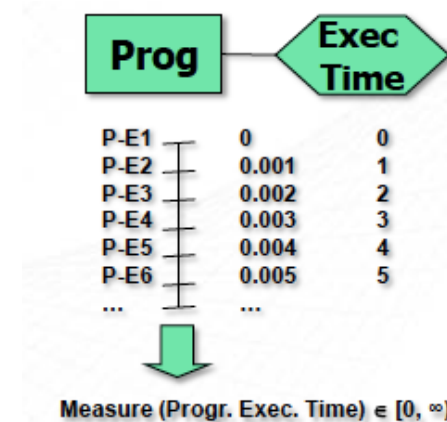
- Los objetos son **jerarquizados** conforme algún criterio.
- Los números que se utilizan para codificar las distintas categorías de una variable sólo permiten establecer relaciones de igualdad/desigualdad y de **orden**.
- **No** se puede precisar la diferencia exacta que existe entre dos objetos.
- Con los “números” de este tipo de variables **no** se pueden efectuar operaciones aritméticas.
- Ejemplo:
 - Medición de complejidad de módulos de software considerando 5 clases de complejidad: trivial, simple, moderada, compleja, incomprensible.

Escala de Intervalo

- Se pueden observar ciertas **diferencias cuantitativas** entre las unidades.
- Se pueden establecer relaciones de igualdad/desigualdad, de orden y además **los intervalos entre los distintos números o valores son iguales**.
- No tiene principio ni final (no existe cero absoluto). Son números arbitrarios en cuanto al origen.
- Con los números de este tipo de variables, se pueden realizar operaciones aritméticas como la suma o la resta, pero no la división ni la multiplicación.
- Ejemplo: *Medición de duración de las distintas fases de un proyecto*
 - *Project scheduling*: Análisis de Requisitos 3 semanas, Diseño 4 semanas, Codificación 4 semanas, Pruebas empieza cuando termina la codificación. ¿Cuándo empieza las pruebas? *Después de 11 semanas*.

Escala de Razón

- Se pueden establecer relaciones de igualdad/desigualdad, de orden, los intervalos entre los distintos valores son iguales y se cuenta con un **verdadero punto cero absoluto**, en relación con el cual se expresan todos los demás valores.
- El valor cero representa el **origen empírico** de la variable, la carencia total de cierta característica.
- Se pueden efectuar todas las operaciones aritméticas (suma, resta, división y multiplicación).
- Ejemplos:
 - Medición del tiempo de ejecución de un programa
 - Líneas de Código (como medidas del tamaño de un programa)



Validación Teórica

- Introducción
- Aproximaciones Axiomáticas
- Aproximaciones basadas en la Teoría de la Medición



Método de Definición

Validación Teórica

- Nos ayuda a saber cuándo y cómo aplicar las métricas.
- Nos proporciona información relativa a las **operaciones realizables con la métrica**.
- Tendencias:
 - Aproximaciones basadas en propiedades
 - Aproximaciones basadas en la teoría de la medición
- Desafortunadamente, no disponemos de un estándar (Van Den Berg y Van Den Broek, 1996).

Método de Definición

Validación Teórica

Aproximaciones Axiomáticas

- Son más sencillas pero de menor utilidad.
- Se limitan a suministrar condiciones necesarias pero no suficientes para los conceptos que definen.
- Sirven para clasificar las métricas por tipos.
- Los marcos formales más conocidos dentro de este tipo son los propuestos por Weyuker (1988), Briand et al. (1996), Morasca y Briand (1997) y Poels and Dedene (1999).

Método de Definición

Validación Teórica

- Marco de Briand et al. (1996) y (1997)

Tamaño
Longitud
Complejidad
Cohesión del sistema
Acoplamiento del sistema



Método de Definición

Validación Teórica

- Un sistema es caracterizado por sus *elementos y relaciones* entre ellos. Un sistema s se representa como un par $\langle E, R \rangle$.
- El **tamaño de un sistema** S es una función $Tamaño(S)$ que queda caracterizada por las siguientes propiedades:
 - **Propiedad 1. No negatividad**
El tamaño de un sistema $S = \langle E, R \rangle$ es no negativo.
 - **Propiedad 2. Valor nulo**
El tamaño de un sistema $S = \langle E, R \rangle$ es nulo si está vacío.
 - **Propiedad 3. Aditividad de módulos**
El tamaño de un sistema $S = \langle E, R \rangle$ es igual a la suma de los tamaños de dos de sus módulos $m_i = \langle E_{m_i}, R_{m_i} \rangle$ y $m_j = \langle E_{m_j}, R_{m_j} \rangle$ tal que cualquier elemento de S es un elemento de m_i o de m_j .

Método de Definición

Validación Teórica

- La **longitud de un sistema*** S es una función $\text{Longitud}(S)$ caracterizada por las siguientes propiedades:
 - **Propiedad 1. No negatividad**
La longitud de un sistema $S = \langle E, R \rangle$ es no negativa
 - **Propiedad 2. Valor nulo**
La longitud de un sistema $S = \langle E, R \rangle$ es nula si E está vacío.

*Métricas de Halstead

Ejemplos de cálculo para las métricas de Halstead:

<http://cnx.org/content/m18034/latest>



UNIVERSIDAD
POLITECNICA
DE VALENCIA

Método de Definición

Validación Teórica

Propiedad 3. Monotonicidad no incremental para componentes conexos.

Sea S un sistema y m un módulo de S tal que m está representado como un componente conexo del grafo de S . La adición de relaciones entre elementos de m no incrementa la longitud de S .

Propiedad 4. Monotonicidad no decreciente para componentes no conexos.

Sea S un sistema y m_1 y m_2 dos módulos de S tal que m_1 y m_2 están representados por dos componentes conectados separados en el grafo que representa S . La adición de relaciones desde elementos de m_1 a elementos de m_2 no decrementa la longitud de S .

Propiedad 5. Módulos disjuntos

La longitud de un sistema $S = \langle E, R \rangle$ compuesto de dos módulos disjuntos m_1 , m_2 es igual al máximo de las longitudes de m_1 y m_2 .

Ejemplo de Métrica de Longitud

Métricas de Halstead

- Su teoría está basada en un conteo (fácil de automatizar) de **número de operadores y operandos en un programa**:
 - los operadores son las palabras reservadas del lenguaje, tales como IF-THEN, READ, FOR,...; los operadores aritméticos +, -, *,..... los de asignación y los operadores lógicos AND, EQUAL TO,....
 - los operandos son las variables, literales y las constantes del programa.
- Halstead distingue entre el # operadores y # operandos únicos y el # total de operadores y operandos. Se utiliza la notación:
 - n1 - número de operadores únicos que aparecen en un programa
 - N1 - número total de ocurrencias de operadores
 - n2 - número de operandos únicos que aparecen en un programa
 - N2 - número total de ocurrencias de operandos

Ejemplo de Métrica de Longitud

Métricas de Halstead

- Las métricas de Halstead para cualquier programa escrito en cualquier lenguaje pueden ser derivadas de estas cuatro cuentas ($n1$, $N1$, $n2$ y $N2$). A partir de ellas han sido elaboradas diferentes medidas para diversas propiedades de los programas: longitud, volumen, etc...
- La **longitud, N , de un programa**, se calcula como:
$$N = N1 + N2$$
- El **volumen, V , de un programa**, se calcula como:
$$V = N \times \log_2(n)$$

donde $n = n1 + n2$

Ejemplo calculo Métricas de Halstead

```
{  
  for (i=2;i<=n;i++)  
    for (j=1;j<=i;j++)  
      if (x[i] < x[j])  
      {  
        aux = x[i];  
        x[i] = x[j];  
        x[j] = aux;  
      }  
}
```

- Operadores:
{..} → 2, for(;;) → 2, = → 5, if → 1, ; → , (..) → 1, < → 1, < = → 2, ++ → 2, [] → 4
- # total de operadores (n1) son 10 y el # de operadores (N1) son 23.
- Operandos:
→ 7, n → 1, j → 6, x → 6, aux → 2
- # total de operandos (n2) son 5 y el # total (N2) son 22.
- La longitud es: $N = 23 + 22 = 45$
- El volumen es: $45 \times \log_2(10+5) = 175, 8$



Método de Definición

Validación Teórica

- La **complejidad** de un sistema S es una función Complejidad(S) que está caracterizada por las siguientes propiedades:
 - Propiedad 1. No negatividad
La complejidad de un sistema $S = \langle E, R \rangle$ es no negativa.
 - Propiedad 2. Valor nulo
La complejidad de un sistema $S = \langle E, R \rangle$ es nula si R está vacío.

Método de Definición

Validación Teórica

Propiedad 3. Simetría

La complejidad de un sistema $S = \langle E, R \rangle$ no depende de la convención elegida para representar las relaciones entre sus elementos.

Propiedad 4. Monotonidad de módulos

La complejidad de un sistema $S = \langle E, R \rangle$ no es menor que la suma de las complejidades de cualquiera dos de sus módulos sin relaciones en común.

Propiedad 5. Aditividad de módulos disjuntos

La complejidad de un sistema $S = \langle E, R \rangle$ compuesto de dos módulos disjuntos m_1 y m_2 es igual a la suma de las complejidades de los dos módulos.

Ejemplo de Métrica de Complejidad

Complejidad ciclomática de McCabe

- Se basa en el recuento del número de caminos lógicos individuales contenidos en un programa. Para calcular la métrica, McCabe utilizó la teoría y flujo de grafos.
- Para hallar la complejidad ciclomática, el programa se representa como un grafo, y cada instrucción que contiene, un nodo del grafo.
- Las posibles vías de ejecución a partir de una instrucción (nodo) se representan en el grafo como aristas.
- Ejemplo:

http://jbravomontero.files.wordpress.com/2012/07/grafica_complejidad_ciclomatica_metodo.png

Ejemplo de Métrica de Complejidad

Complejidad ciclomática de McCabe

```
1 if (condicion){
2   if (condicion){
3     A;
4     B;
5   } else {
6     C;
7     D;
8   }
9 }
```

- Si se realizase el grafo, se observaría que se encuentran 3 caminos posibles para llegar de la sentencia 1 a la sentencia 6:
 - Camino 1 (si ambos IF's son verdad): Sentencias 1, 2, 3, 6
 - Camino 2 (si el primer IF es verdad y el segundo es falso): Sentencias 1, 4, 6
 - Camino 3 (si el primer IF es falso): Sentencias 1, 6
- Este programa tiene una complejidad ciclomática de 3.
- La complejidad ciclomática se puede calcular de otras maneras, e.j., con la fórmula:

$$v(G) = e - n + 2$$

donde e representa el #de aristas y n el #de nodos.



Método de Definición

Validación Teórica

La **cohesión*** de un módulo $m = \langle E_m, R_m \rangle$ de un sistema modular MS es una función $\text{Cohesión}(m)$ caracterizada por las siguientes propiedades.

Propiedad 1. No negatividad y normalización

La cohesión de un módulo $m = \langle E_m, R_m \rangle$ de un sistema modular $MS = \langle E, R, M \rangle$ pertenece a un intervalo especificado.

Propiedad 2. Valor nulo

La cohesión de un módulo $m = \langle E_m, R_m \rangle$ de un sistema modular $MS = \langle E, R, M \rangle$ es nula si R_m está vacío.

*Se refiere al grado de **"adhesivo" interno** con el que se ha construido el módulo/componente. Un módulo es cohesivo si todos sus elementos están orientados a la realización de una única tarea y son esenciales para llevarla a cabo.

Método de Definición

Validación Teórica

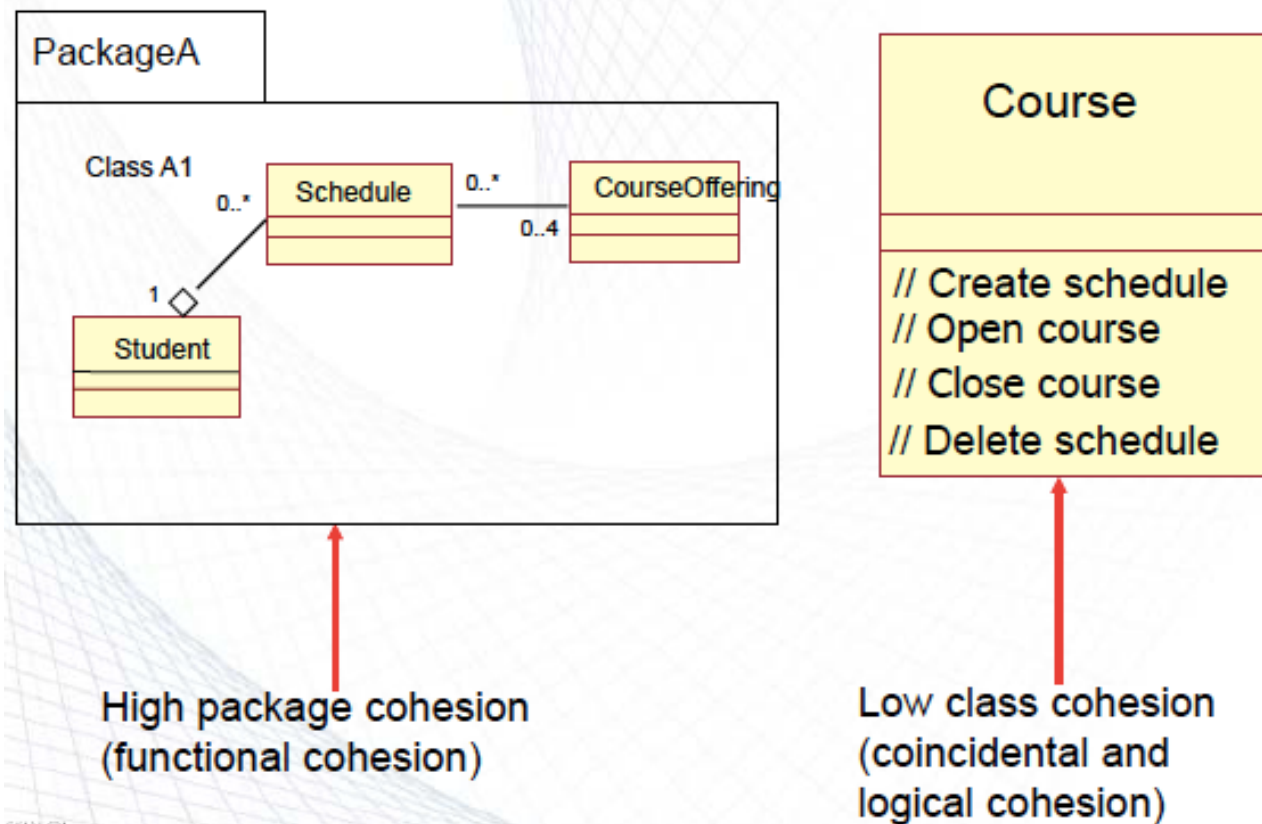
Propiedad 3. Monotonicidad

Sea $MS' = \langle E, R', M' \rangle$ y $MS'' = \langle E, R'', M'' \rangle$ dos sistemas modulares (con el mismo conjunto de elementos E) tal que existen dos módulos $m' = \langle E_m, R'_m \rangle$ y $m'' = \langle E_m, R''_m \rangle$ (con el mismo conjunto de elementos E_m) pertenecientes a M' y M'' respectivamente, tal que $R' - R'_m = R'' - R''_m$, y $R_m \subseteq R_{m''}$ (que implica que $IR' \subseteq IR''$). Entonces, $\text{Cohesión}(m') \leq \text{Cohesión}(m'')$

Propiedad 4. Módulos con cohesión

Sean $MS' = \langle E, R, M' \rangle$ y $MS'' = \langle E, R, M'' \rangle$ dos sistemas modulares (con el mismo sistema subyacente $\langle E, R \rangle$) tal que $M'' = M' - \{m'_1, m'_2\} \cup \{m''\}$, con $m'_1 \in M'$, $m'_2 \in M'$, $m'' \notin M'$, y $m'' = m'_1 \cup m'_2$. (Los dos módulos m'_1 y m'_2 son sustituidos por el módulo m'' , unión de m'_1 y m'_2). Si no existen relaciones entre elementos pertenecientes a m'_1 y m'_2 , esto es, $\text{InputR}(m'_1) \cap \text{OutputR}(m'_2) = \emptyset$ e $\text{InputR}(m'_2) \cap \text{OutputR}(m'_1) = \emptyset$, entonces $\max\{\text{Cohesión}(m'_1), \text{Cohesión}(m'_2)\} \geq \text{Cohesión}(m'')$

Ejemplos de Cohesión



*Se refiere al grado de **"adhesivo" interno** con el que se ha construido el módulo/componente. Un módulo es cohesivo si todos sus elementos están orientados a la realización de una única tarea y son esenciales para llevarla a cabo.



Método de Definición

Validación Teórica

El **acoplamiento*** de un módulo $m = \langle E_m, R_m \rangle$ de un sistema modular MS es una función $\text{Acoplamiento}(m)$ caracterizada por las siguientes propiedades:

Propiedad 1. No negatividad

El acoplamiento de un módulo $m = \langle E_m, R_m \rangle$ de un sistema modular es no negativo.

Propiedad 2. Valor nulo.

El acoplamiento de un módulo $m = \langle E_m, R_m \rangle$ de un sistema modular es nulo si $\text{OuterR}(m)$ está vacío.

Propiedad 3. Monotonicidad.

Sean $MS' = \langle E, R', M' \rangle$ y $MS'' = \langle E, R'', M'' \rangle$ dos sistemas modulares (con el mismo conjunto de elementos E) tal que existen dos módulos $m' \in M'$, $m'' \in M''$ tal que $R' - \text{OuterR}(m') = R'' - \text{OuterR}(m'')$, y $\text{OuterR}(m') \subseteq \text{OuterR}(m'')$. Entonces, $\text{Acoplamiento}(m') \leq \text{Acoplamiento}(m'')$

*Se dice que dos módulos, componentes (o clases) están **altamente acoplados** cuando existe mucha dependencia entre ellos. Los módulos **poco acoplados**, tienen algunas dependencias, pero son débiles.

Método de Definición

Validación Teórica

Propiedad 4. Fusión de módulos

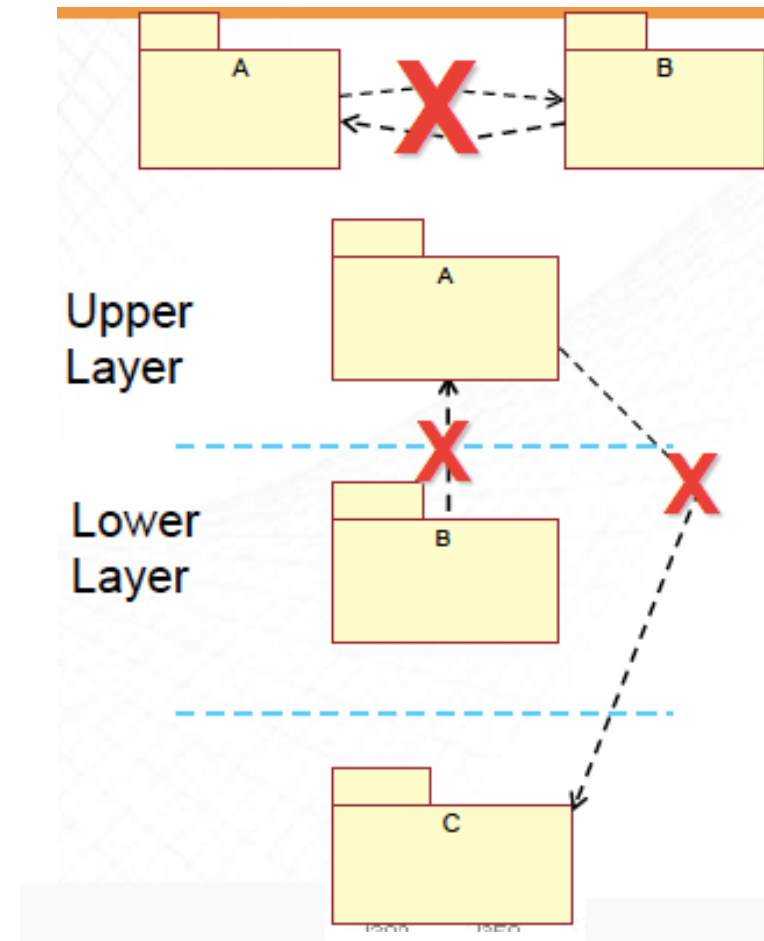
Sean $MS' = \langle E', R', M' \rangle$ y $MS'' = \langle E'', R'', M'' \rangle$ dos sistemas modulares tales que $E' = E''$, $R' = R''$, y $M'' = M' - \{m'_{1}, m'_{2}\} \cup \{m''\}$, donde $m'_{1} = \langle E_{m'_{1}}, R_{m'_{1}} \rangle$, $m'_{2} = \langle E_{m'_{2}}, R_{m'_{2}} \rangle$, y $m'' = \langle E_{m''}, R_{m''} \rangle$ con $m'_{1} \in M'$, $m'_{2} \in M'$, $m'' \notin M'$, y $E_{m''} = E_{m'_{1}} \cup E_{m'_{2}}$ y $R_{m''} = R_{m'_{1}} \cup R_{m'_{2}}$. (Los dos módulos m'_{1} y m'_{2} son sustituidos por el módulo m'' , cuyos elementos y relaciones son la unión de los de m'_{1} y m'_{2}). Entonces $\text{Acoplamiento}(m'_{1}) + \text{Acoplamiento}(m'_{2}) \geq \text{Acoplamiento}(m'')$

Propiedad 5. Aditividad de módulos disjuntos

Sean $MS' = \langle E, R, M' \rangle$ y $MS'' = \langle E, R, M'' \rangle$ dos sistemas modulares (con el mismo sistema subyacente $\langle E, R \rangle$) tal que $M'' = M' - \{m'_{1}, m'_{2}\} \cup \{m''\}$, con $m'_{1} \in M'$, $m'_{2} \in M'$, $m'' \notin M'$, y $m'' = m'_{1} \cup m'_{2}$. Si no existen relaciones entre elementos pertenecientes a m'_{1} y m'_{2} , esto es, $\text{InputR}(m'_{1}) \cap \text{OutputR}(m'_{2}) = \emptyset$ e $\text{InputR}(m'_{2}) \cap \text{OutputR}(m'_{1}) = \emptyset$, entonces $\text{Acoplamiento}(m'_{1}) + \text{Acoplamiento}(m'_{2}) = \text{Acoplamiento}(m'')$

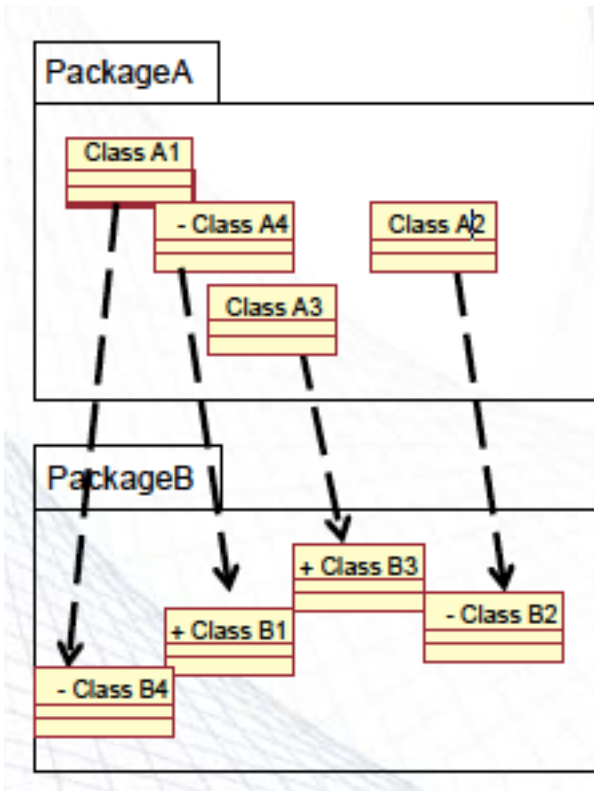
Ejemplos de Acoplamiento

- Acoplamiento describe la fuerza con un elemento se relaciona con otro elemento.
- El objetivo es lograr “bajo acoplamiento”.
- Ejemplos de alto acoplamiento en el diseño OO:
 - Paquetes no deben tener dependencias entre sí.
 - Paquetes en capas más bajas no deben depender de paquetes que están en capas más altas.
 - Por lo general, las dependencias no deben saltar capas (al menos que se haya especificado en la arquitectura del sistema).

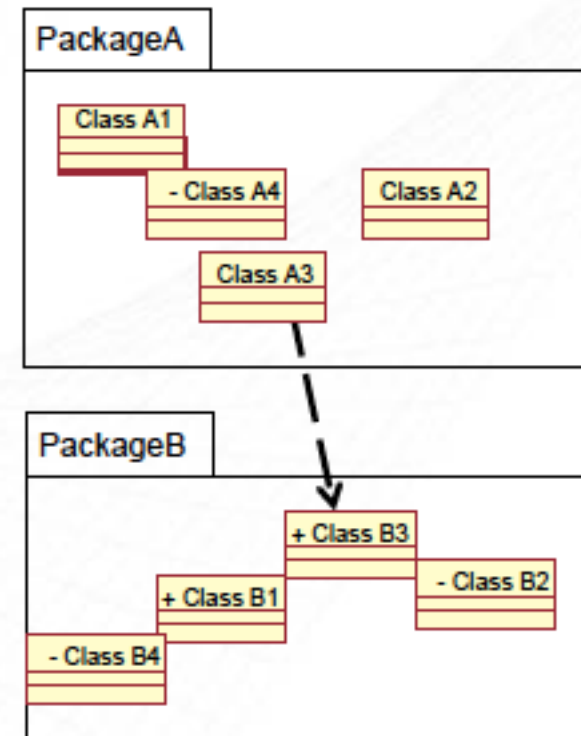


Ejemplos de Acoplamiento

- Intente alcanzar el acoplamiento más bajo.



Alto acoplamiento



Bajo acoplamiento



Método de Definición

Validación Teórica

Aproximaciones basadas en la Teoría de la Medición

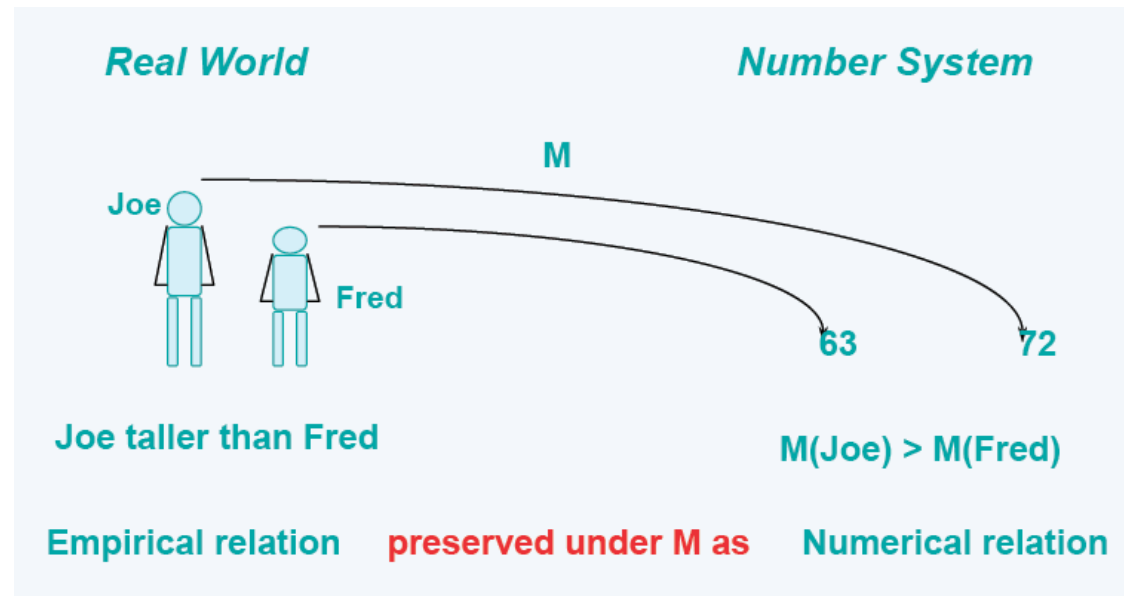
- Aparato matemático más complejo
- Su objetivo es obtener la escala matemática a la que pertenece una métrica, y por tanto sus transformaciones admisibles, estadísticos y tests estadísticos aplicables.
- Se logra extraer más información al validar una métrica en un marco formal de este tipo.
- Los marcos formales más conocidos son los propuestos por Zuse (1998) o Whitmire (1997).

Método de Definición

Validación Teórica

Aproximaciones basadas en la Teoría de la Medición

- Fundamentalmente, en la teoría de la medición se identifica tres componentes principales: un sistema relacional empírico, un sistema relacional numérico y la correspondencia (*mapping*) entre ambos sistemas relacionales.



Método de Definición

Validación Teórica

■ Marco de Zuse (1998)

- En el área de la ingeniería del software existen muchas posibilidades de concatenar los objetos de interés (módulos de programas, clases del modelo de objetos, etc.).
- Tales **operaciones de concatenación** permiten definir diversas estructuras de medición.
- Zuse se basa en las estructuras extensivas, a continuación introducimos este tipo de estructuras.

Método de Definición

Validación Teórica

■ Marco de Zuse (1998)

Estructura extensiva modificada

Sea A un conjunto no vacío, $\bullet \geq$ una relación binaria en A y \circ una operación binaria cerrada sobre A . El sistema relacional $(A, \bullet \geq, \circ)$ es una estructura modificada extensiva si y sólo si los siguientes axiomas se cumplen para todos $A_1, \dots, A_4 \in A$.

- Axioma1: $(A, \bullet \geq)$ (orden débil)
- Axioma2: $A_1 \circ A_2 \bullet \geq A_1$ (positividad)
- Axioma3: $A_1 \circ (A_2 \circ A_3) \approx (A_1 \circ A_2) \circ A_3$ (asociatividad débil)
- Axioma4: $A_1 \circ A_2 \approx A_2 \circ A_1$ (conmutatividad débil)
- Axioma5: $A_1 \bullet \geq A_2 \Rightarrow A_1 \circ A \bullet \geq A_2 \circ A$ (monotonidad débil)
- Axioma6: Si $A_3 \bullet > A_4$ entonces para cualquier A_1, A_2 , existe un número natural n , tal que $A_1 \circ nA_3 \bullet > A_2 \circ nA_4$ (axioma arquimedeano)

Método de Definición

Validación Teórica

Reglas de combinación

Si una medida u está en una escala ordinal $((A, \bullet \geq), (\mathfrak{R}, \geq), u)$ entonces, una regla de combinación f se define como: $\mu(A1 \text{ o } A2) = f(\mu(A1), \mu(A2))$, donde $A1, A2 \in A$ es un conjunto de objetos, o es una operación binaria (operación de concatenación) y u es una medida.

condiciones de independencia

C1: $A1 \approx A2 \Rightarrow A1 \text{ o } A3 \approx A2 \text{ o } A3$ y $A1 \approx A2 \Rightarrow A3 \text{ o } A1 \approx A3 \text{ o } A2$

C2: $A1 \approx A2 \Leftrightarrow A1 \text{ o } A3 \approx A2 \text{ o } A3$ y $A1 \approx A2 \Leftrightarrow A3 \text{ o } A1 \approx A3 \text{ o } A2$

C3: $A1 \bullet \geq A2 \Rightarrow A1 \text{ o } A3 \bullet \geq A2 \text{ o } A3$, y $A1 \bullet \geq A2 \Rightarrow A3 \text{ o } A1 \bullet \geq A3 \text{ o } A2$

C4: $A1 \bullet \geq A2 \Leftrightarrow A1 \text{ o } A3 \bullet \geq A2 \text{ o } A3$, y $A1 \bullet \geq A2 \Leftrightarrow A3 \text{ o } A1 \bullet \geq A3 \text{ o } A2$

Método de Definición

Validación Teórica

Estructura modificada de creencia

MRB1: $\forall A, B \in \mathfrak{S}: A \bullet \geq B \text{ o } B \bullet \geq A$ (compleción)

MRB2: $\forall A, B, C \in \mathfrak{S}: A \bullet \geq B \text{ y } B \bullet \geq C \Rightarrow A \bullet \geq C$
(transitividad)

MRB3: $\forall A \subseteq B \Rightarrow A \bullet \geq B$ (axioma de dominancia)

MRB4: $\forall (A \supset B, A \cap C = \phi) \Rightarrow (A \bullet \geq B \Rightarrow A \cup C \bullet > B \cup C)$
(monotonicidad parcial)

MRB5: $\forall A \in \mathfrak{S}: A \bullet \geq 0$ (positividad)

Método de Definición

Validación Teórica

- Si una métrica cumple el *orden débil*, puede ser clasificada en la **escala ordinal**.
- Si una métrica cumple la *estructura modificada extensiva* y además cumple *las condiciones de independencia* entonces podrá ser clasificada dentro de la **escala de ratio**.
- Si una métrica *no satisface la estructura extensiva* pero si las *condiciones de independencia* puede ser clasificada dentro de la **escala ordinal**.
- Si una métrica cumple la *estructura modificada de creencia*, podrá ser caracterizada **por encima de la escala ordinal** pero sin llegar a la escala de ratio.

Validación Empírica

- Métodos Experimentales
- Ejemplo de Validación Empírica de Métricas



Método de Definición

Validación Empírica

- Se utiliza para obtener información objetiva sobre la utilidad de las métricas propuestas
- El estudio empírico resulta necesario para comprobar y entender las implicaciones de las medidas de nuestros productos
- Esto se consigue a través de hipótesis en el mundo real que habrá que comprobar con datos empíricos

Método de Definición

Validación Empírica

Kish (1959) divide las investigaciones empíricas en:

- **Experimentos:** son las investigaciones en las que las posibles variables perturbadoras han sido aleatorizadas.
- **Casos de estudio:** son aquellos en los que no hay aleatoriedad de variables perturbadoras ni representatividad de los sujetos que componen la muestra de estudio
- **Encuestas (Surveys):** son investigaciones en las que los sujetos del estudio son una muestra representativa de la población a la que pertenecen.

Método de Definición

Validación Empírica

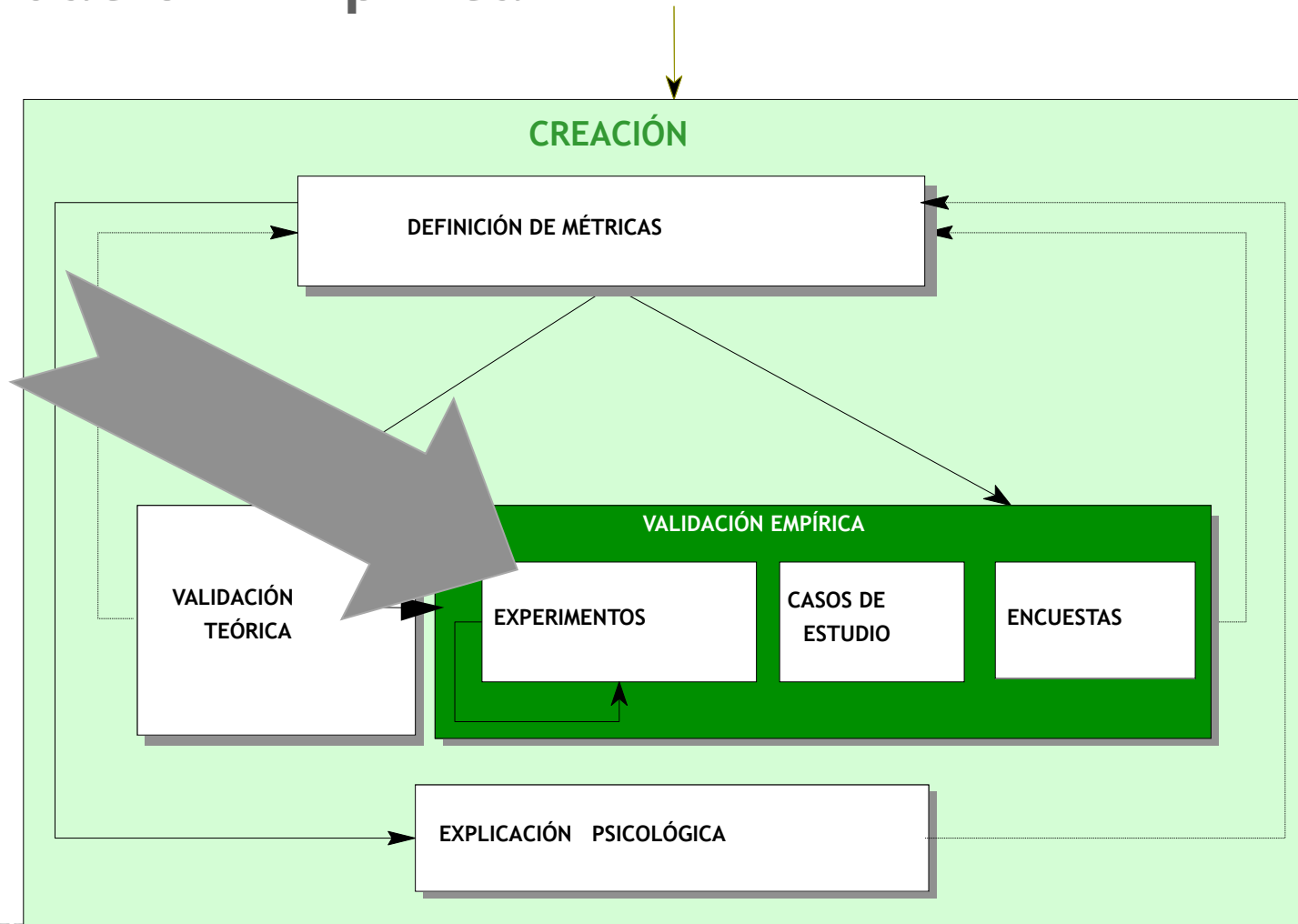
Experimentos vs. Casos de Estudio

Factor	Experimentos	Casos de estudio
Nivel de control	Alto	Bajo
Dificultad de controlar	Baja	Alta
Nivel de réplica	Alto	Bajo
Coste de replicar	Bajo	Alto



Método de Definición

Validación Empírica



Método de Definición

Validación Empírica

- Fases fundamentales:
 - Determinación del problema
 - Creación de la hipótesis
 - Comprobación de la hipótesis
 - Análisis de resultados

Método de Definición

Validación Empírica

Determinación del problema

El método experimental sólo puede ser utilizado en aquellos casos que pueden ser resolubles en términos de causalidad.

Además, la exigencia de que la situación está totalmente controlada, suele forzar al investigador a tratar con aspectos muy reducidos.

Otra característica que hay que tener en cuenta es que, tal y como indican Basili et al. (1999), las réplicas de los experimentos son necesarias.

Método de Definición

Validación Empírica

Creación de la hipótesis

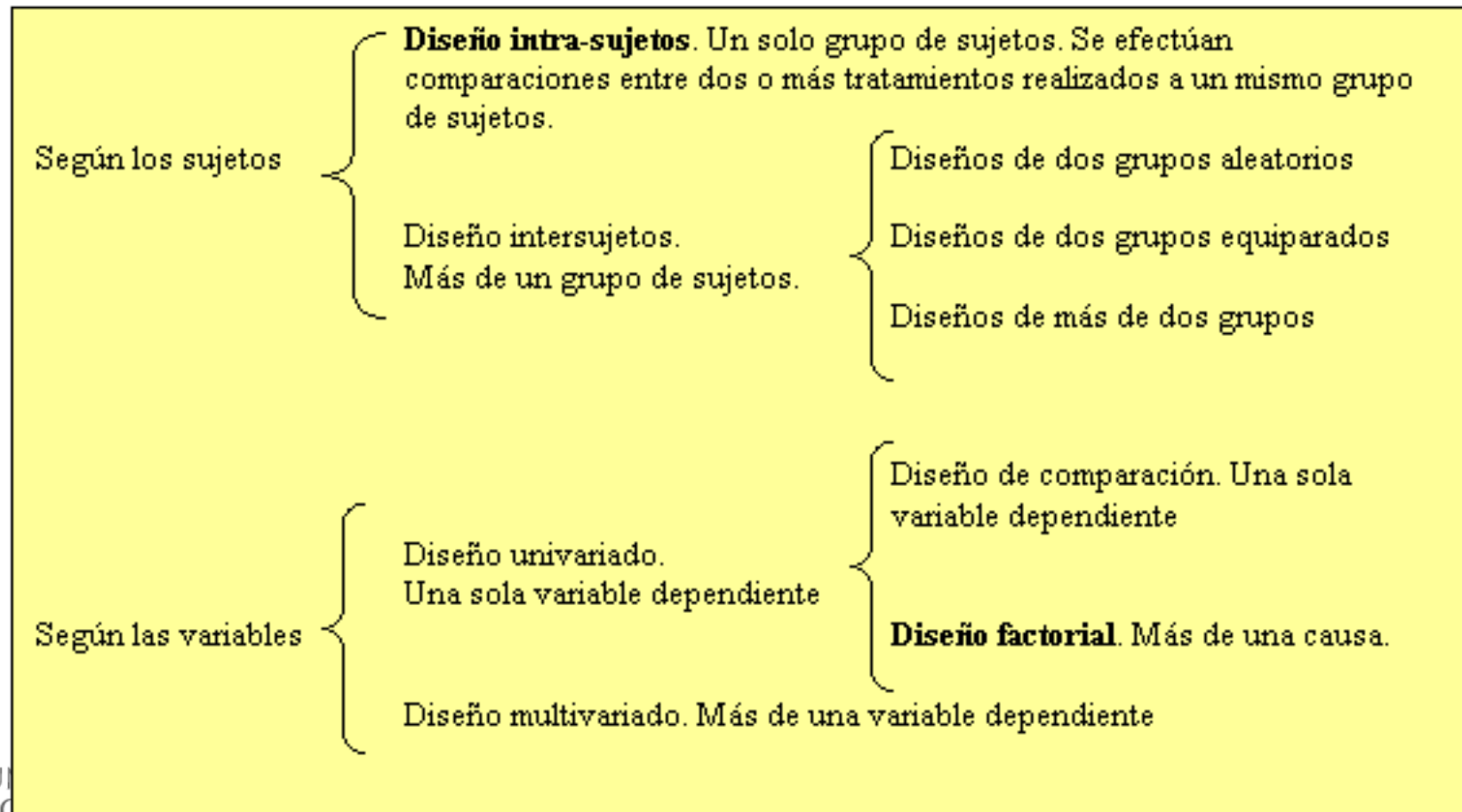
Con la definición del problema en “lenguaje natural” no se puede pasar directamente a su comprobación empírica por lo que se definen las hipótesis de trabajo.

A partir de la hipótesis de trabajo, hay que montar el experimento, especificando las condiciones concretas y controladas en las que se va a poner a prueba la hipótesis (diseño).

Método de Definición

Validación Empírica

Diseño



Método de Definición

Validación Empírica

Comprobación de la hipótesis

Ponemos en práctica la situación, es decir, hay que realizar el experimento.

Esta realización ha de ajustarse a lo previsto en el diseño.

Por eso, es conveniente planificar el experimento, siendo conveniente llevar a cabo un experimento piloto.

Método de Definición

Validación Empírica

Análisis de resultados

Tras realizar el experimento, obtenemos una serie de datos.

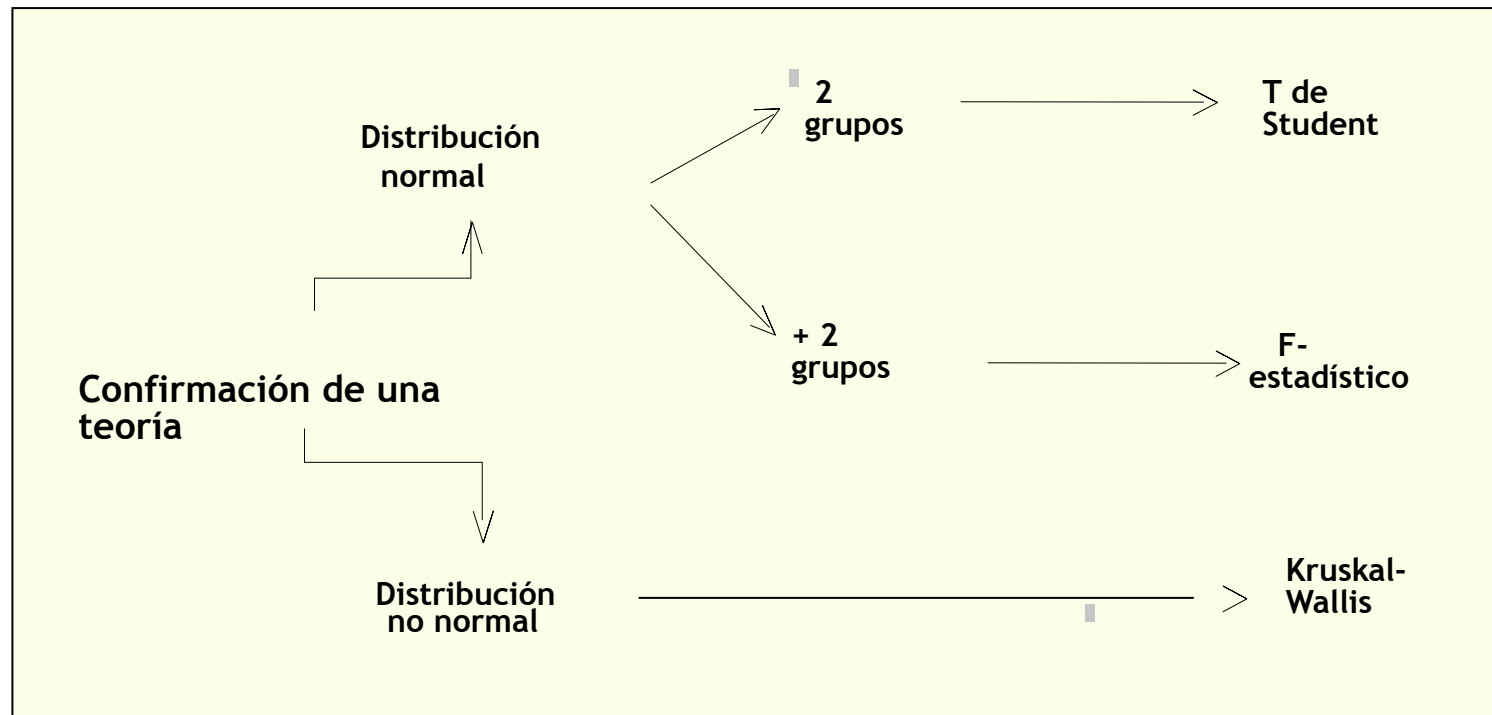
Estos datos no son directamente interpretables y deben ser sometidos a ciertas operaciones estadísticas a partir de las cuales obtenemos los resultados del experimento.

A esta labor estadística se le denomina análisis de los datos.

Método de Definición

Validación Empírica

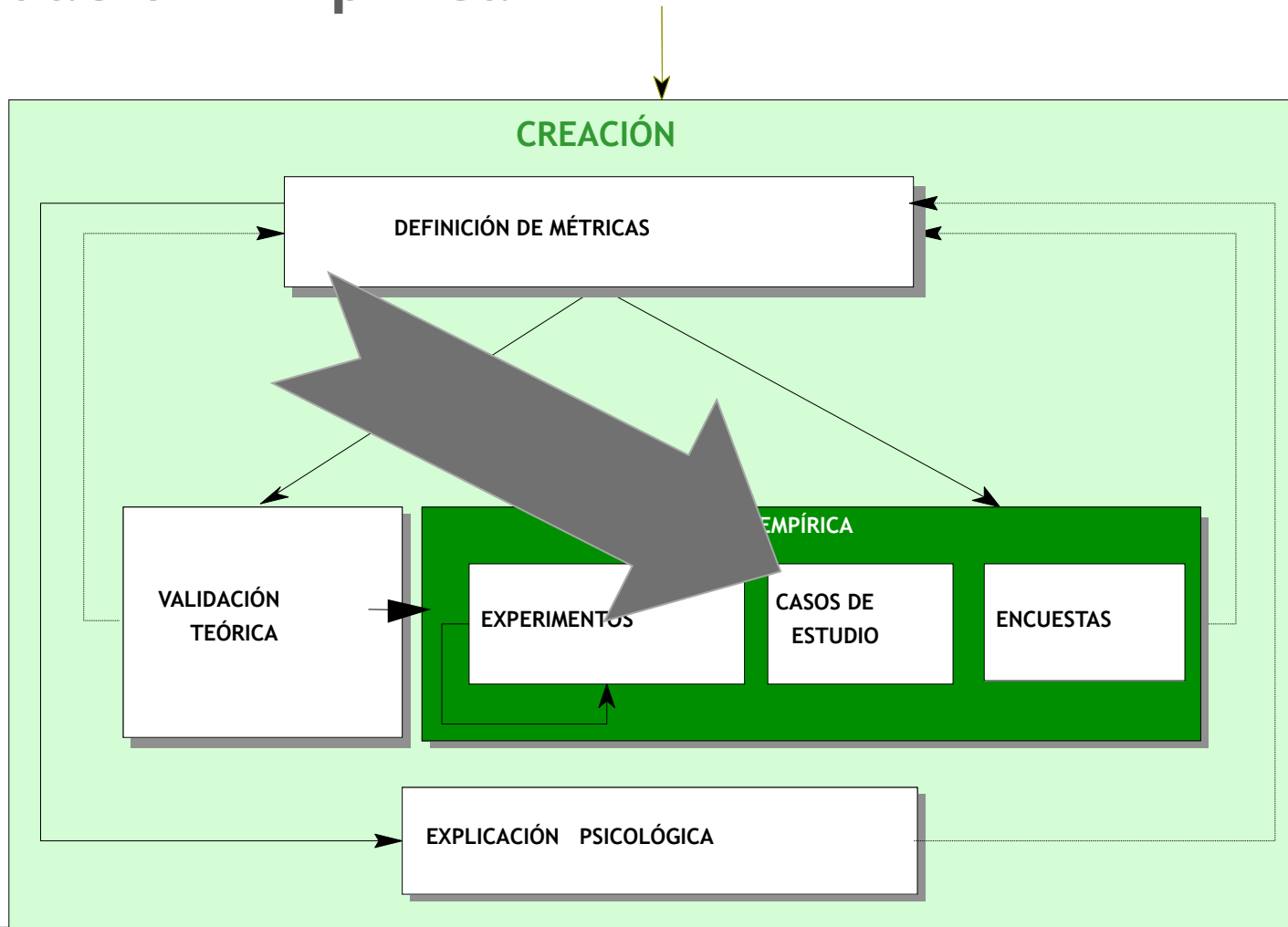
Pfleeger (1995)



UNIVERSIDAD
POLITECNICA
DE VALENCIA

Método de Definición

Validación Empírica



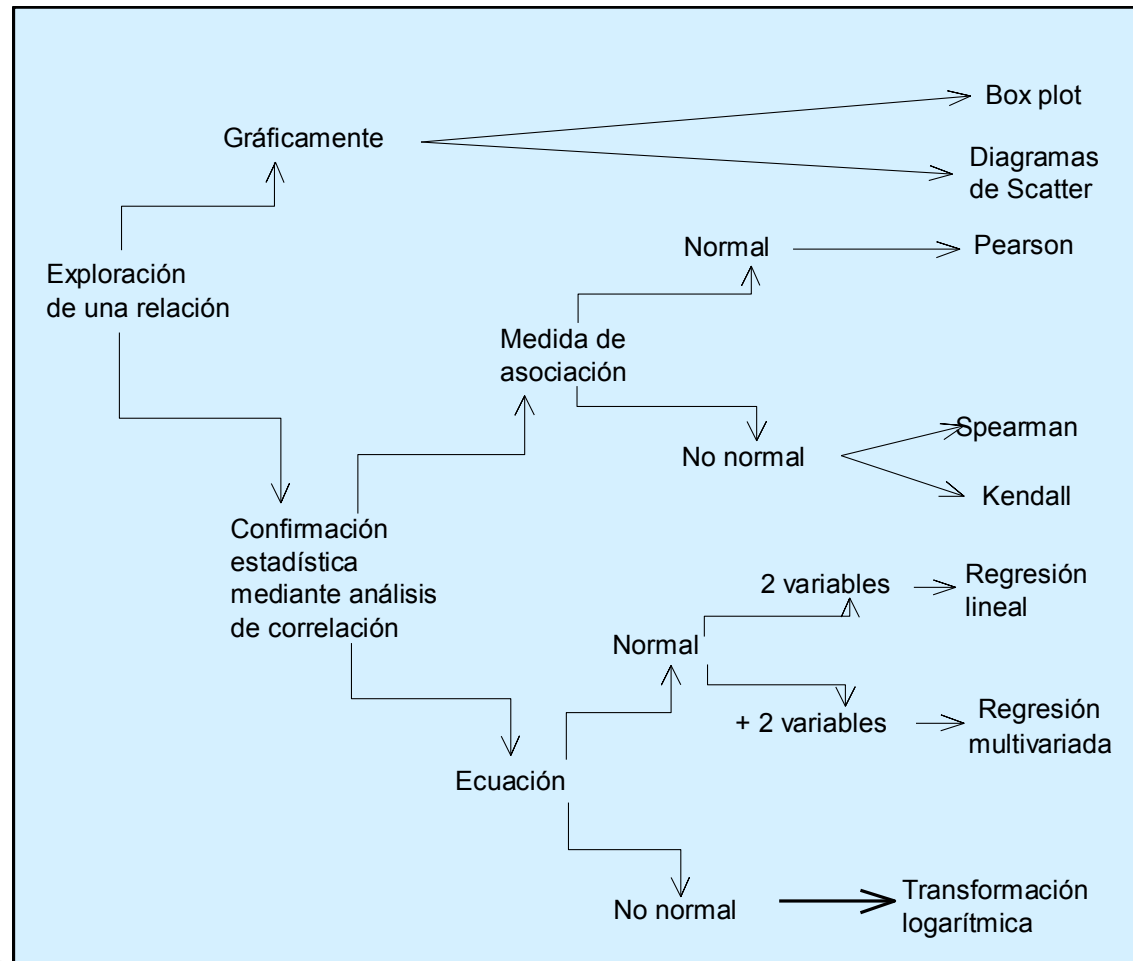
Método de Definición

Validación Empírica

- Hay ocasiones en las que el investigador se dedica a observar o registrar lo que sucede en una situación natural.
- No introduce ninguna variable para comprobar si ejerce efectos sobre la conducta de los sujetos ni asigna aleatoriamente los sujetos del estudio a grupos distintos. Simplemente observa.
- El caso de estudio complementa al experimento.

Método de Definición

Validación Empírica



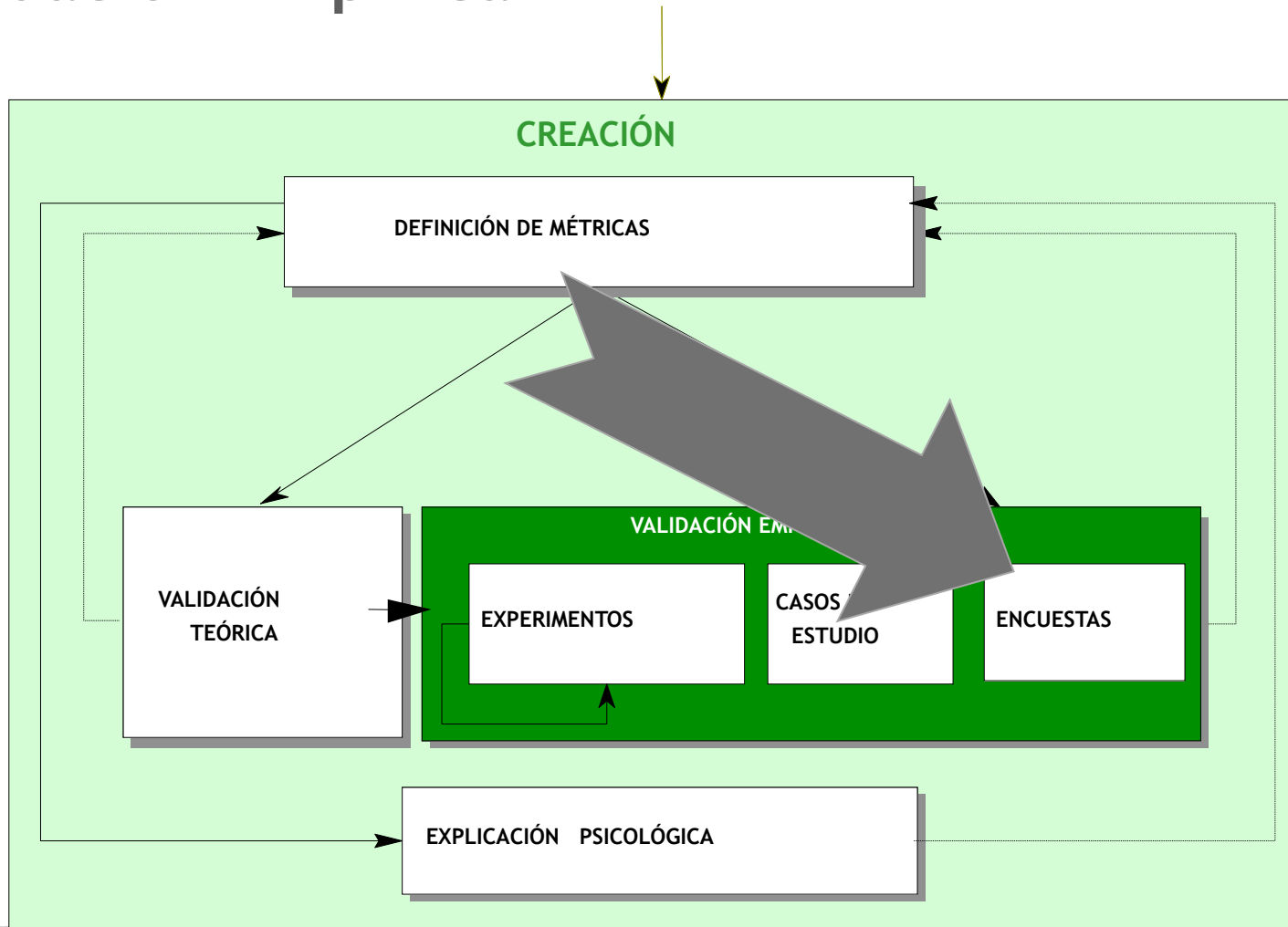
Método de Definición

Validación Empírica

- Técnicas estadísticas tradicionales
- Técnicas avanzadas de análisis de datos:
 - Aprendizaje automático
 - RoC
 - C4.5

Método de Definición

Validación Empírica



Método de Definición

Validación Empírica

- Una encuesta es un sistema completo para:
 - recoger información,
 - describir,
 - comparar
 - o explicar
- conocimiento, actitudes y comportamiento
- El primer paso antes de comenzar cualquier encuesta es fijar los objetivos con los resultados que se esperan de la encuesta.
- Una vez hecho esto, el siguiente paso a realizar consiste en diseñar el cuestionario.

Método de Definición

Validación Empírica

- El diseño ha de ser adecuado para conseguir nuestros objetivos:
 - Debe evitar parcialidad: Los resultados de la encuesta reflejen la realidad de la situación.
 - Debe ser apropiada: debe tener sentido dentro del contexto de la población.
 - Debe tener un coste efectivo: se quiere un diseño cuya distribución y análisis estén dentro de los recursos destinados a la encuesta.

Método de Definición

Validación Empírica

- Para construir un instrumento de encuesta hay dos opciones:
 - Buscar en la literatura
 - Construir un instrumento
- Cuando los investigadores utilizan instrumentos existentes tienen las siguientes ventajas:
 - Se ha evaluado la validez y la fiabilidad
 - Se pueden comparar los nuevos resultados con los resultados de otros estudios.
- Pero, en ocasiones ocurre que no es posible utilizar de forma directa un instrumento existente, por lo que se ha de modificar

Método de Definición

Validación Empírica

- La siguiente tarea a realizar consiste en la evaluación del cuestionario. Para ello, se ha de tener en cuenta:
 - Cómo motivar a las personas que responden
 - Cómo evitar o disminuir la parcialidad de la persona que responde.
 - Cómo evaluar los cuestionarios y los instrumentos de encuesta.

Método de Definición

Validación Empírica

- También es necesario determinar la muestra.
- Para obtener una muestra, se debe comenzar por definir una población destinataria.
- Los métodos de muestreo son: probabilísticos (Muestra aleatoria simple, Muestra aleatoria estratificada, Muestreo sistemático, Muestreo basado en grupos) o no probabilísticos (Muestreo por conveniencia, Muestreo que aumenta progresivamente, Muestreo de cuota)

Método de Definición

Validación Empírica

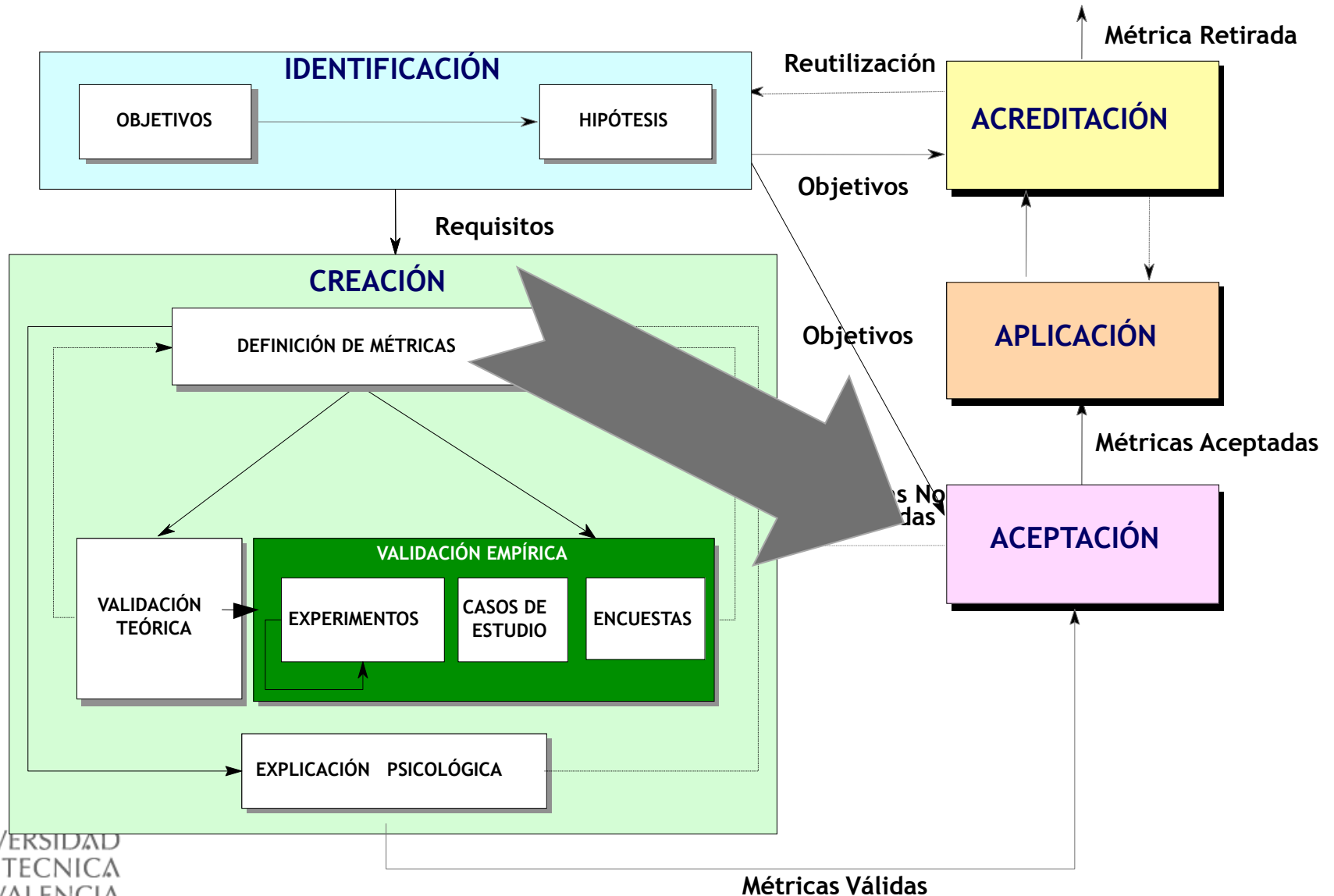
- Por regla general, se realiza la encuesta a la muestra anteriormente obtenida, ya que las respuestas de un grupo pequeño deben representar lo que hubiese ocurrido si se hubiese entrevistado a todo el grupo entero.

Método de Definición

Validación Empírica

- A partir de ahí se aplica el cuestionario y
- Se analizan los resultados para ver si se cumplen las hipótesis de partida

Método de Definición de Métricas



Método de Definición

Aceptación

- Una vez obtenida una métrica válida, suele ser necesario pasar por una etapa de aceptación de la métrica en la que se harán pruebas en entornos reales, de manera que podamos comprobar si la métrica cumple los objetivos deseados dentro del campo de aplicación real.

Método de Definición

Aplicación y Acreditación

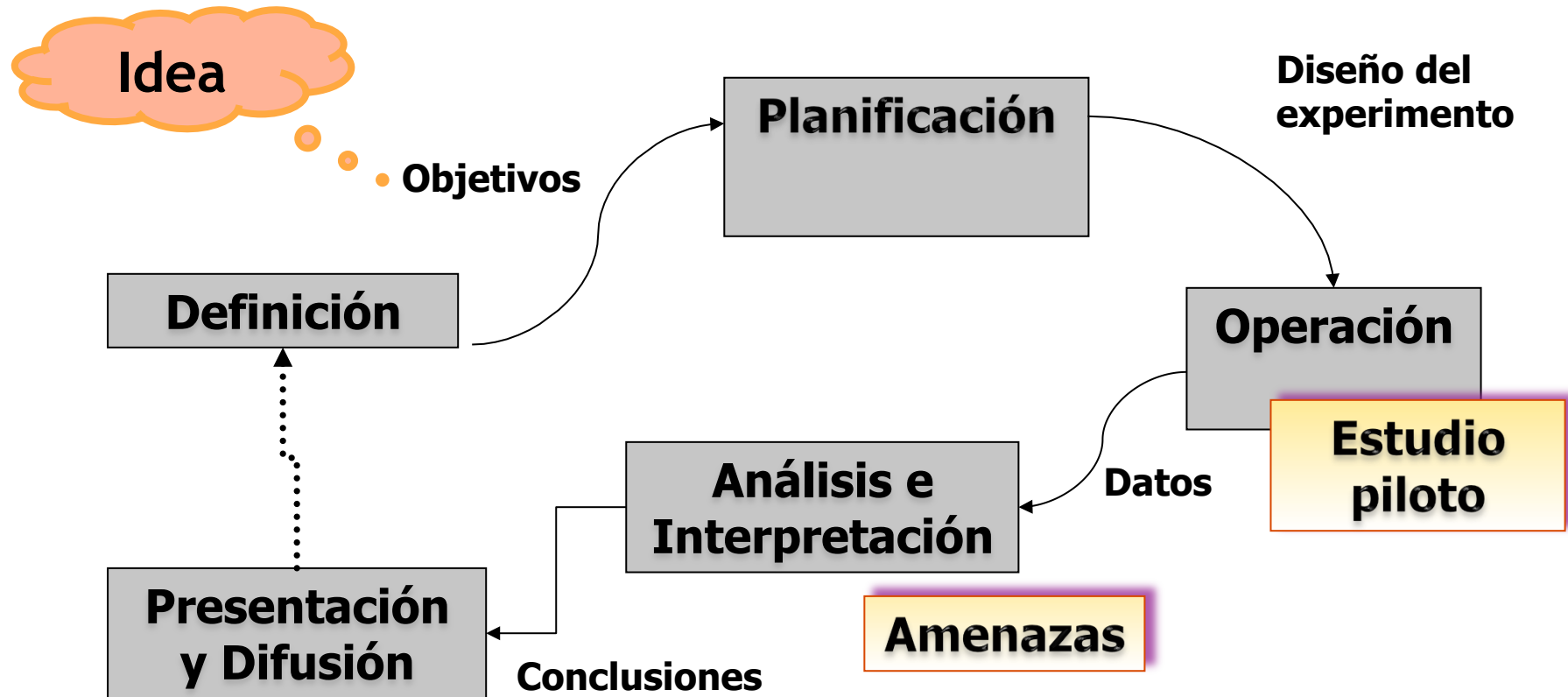
- **Aplicación.** Una vez que tengamos una métrica aceptada, la utilizaremos dentro del campo de aplicación para la que fue definida.
- **Acreditación.** Es la última etapa del método, que discurre en paralelo con la fase de aplicación y tiene como objetivo el **mantenimiento de la métrica**, de manera que se pueda adaptar al entorno cambiante. Como consecuencia de esta etapa, puede que una métrica sea retirada, porque ya no sea útil en el entorno en el que se aplica, o que sea reutilizada para iniciar el proceso de nuevo.

Ejemplo Validación Empírica Métricas

- El **objetivo** del estudio empírico es:

Determinar si dos métricas, *NFK (Number of Foreign Keys)* y *DRT (Depth of Referential Tree)* pueden ser utilizadas como mecanismos para controlar la mantenibilidad de las BD relacionales.

Proceso Experimental



Wohlin C., Runeson P., Höst M., Ohlson M., Regnell B. y Wesslén A.
(2000) Experimentation in Software Engineering: An Introduction,
Kluwer Academic Publishers.

Ejemplo

Definición

- La definición del **objetivo del experimento** puede ser definido como:

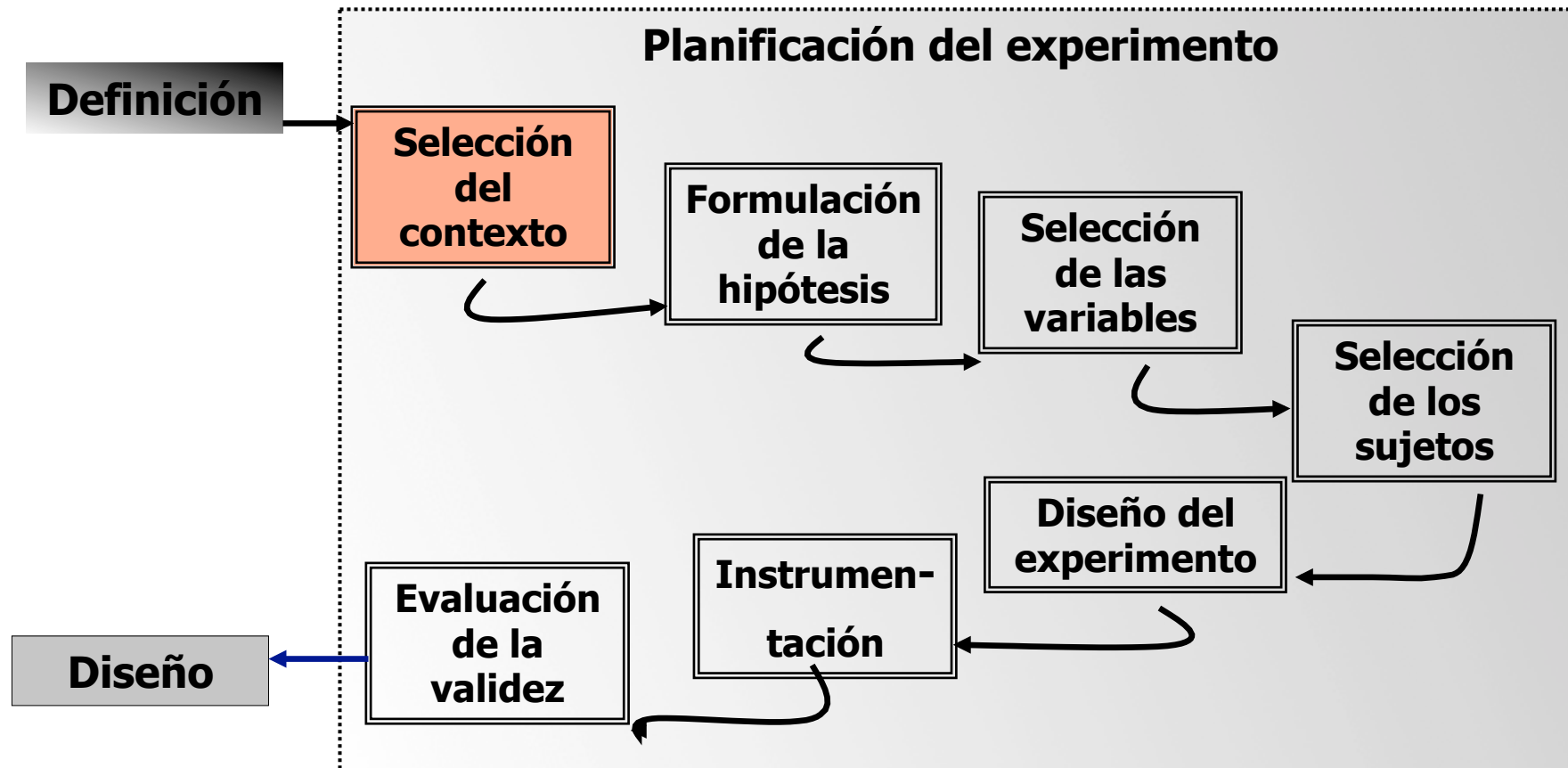
Analizar las métricas para bases de datos relacionales con el **propósito** de evaluar si pueden ser utilizados como mecanismos útiles

respecto a la analizabilidad de las BD relacionales desde el **punto de vista** de los investigadores

en el contexto de profesionales con experiencia en bases de datos

Ejemplo

Planificación del Experimento



Ejemplo

Planificación del Experimento

Selección del contexto

- Caracterización: **más de un objeto** y **más de un sujeto**
- El contexto del experimento es un grupo de profesionales y se lleva cabo “**off-line**”
- Los sujetos son **profesionales**
- El experimento aborda un **problema real**
- El experimento es **específico**

Ejemplo

Planificación del Experimento

Formulación de la hipótesis

- Las hipótesis del experimento son:
 - **Hipótesis Nula:** Valores diferentes de DRT y NFK no afectan a la analizabilidad del esquema de la base de datos.
 - **Hipótesis alternativa H1:** El valor de DRT afecta a la analizabilidad del esquema de la base de datos.
 - **Hipótesis alternativa H2:** El valor de NFK afecta a la analizabilidad del esquema de la base de datos.
 - **Hipótesis alternativa H3:** La interacción de DRT y NFK afecta a la analizabilidad del esquema de la base de datos.

Ejemplo

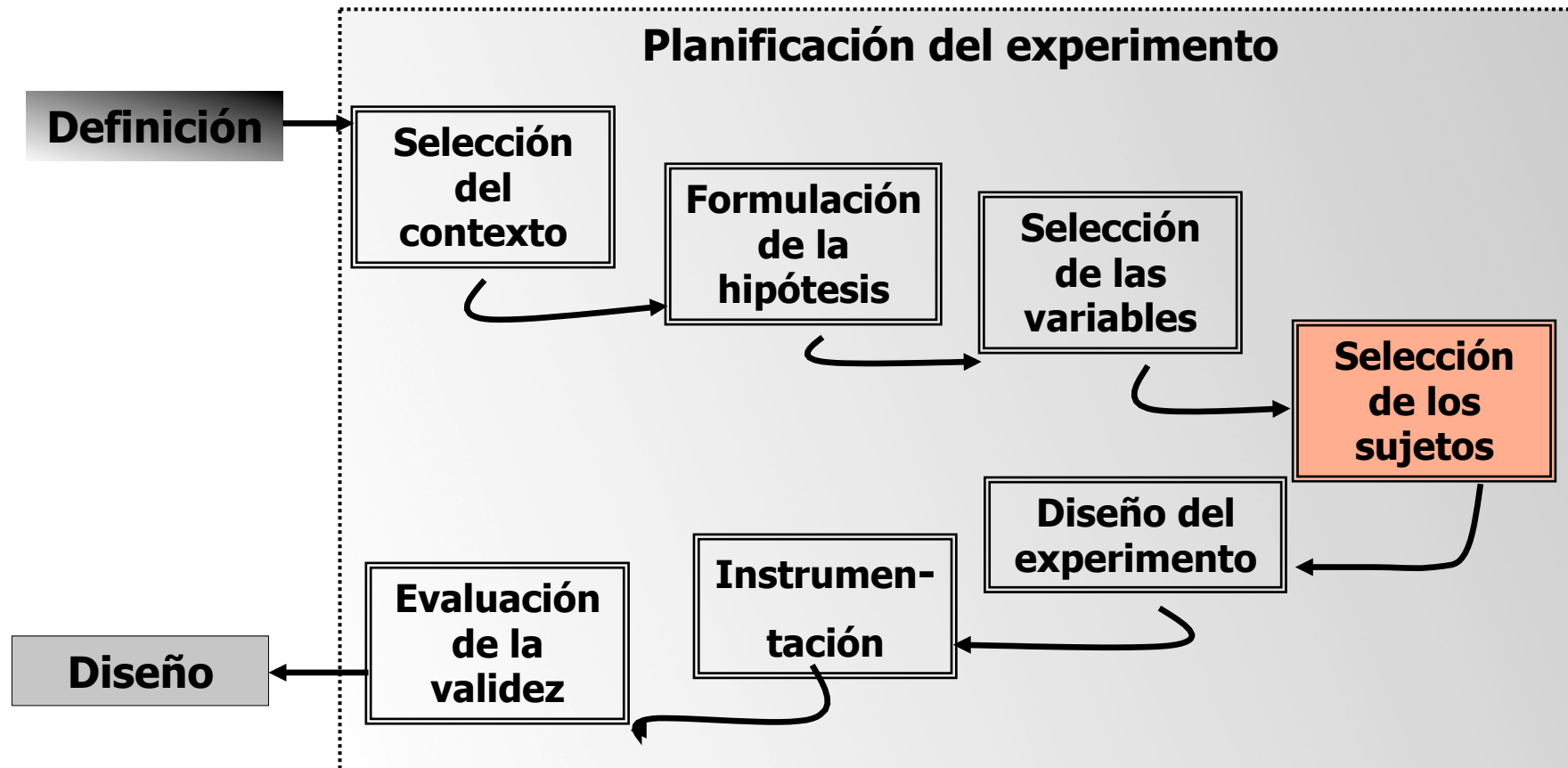
Planificación del Experimento

Selección de las variables

- **Variables Independientes.** Se corresponden con las dos métricas bajo estudio: NFK y DRT. Cada una de estas métricas (factores) pueden tomar dos valores diferentes (niveles): 5 y 8 para la métrica NFK y 2 y 5 para la DRT.
- **Variables Dependientes.** La analizabilidad se midió como el tiempo que cada sujeto utilizó para realizar todas las tareas de cada test experimental y la corrección con la que completaban las tareas incluidas en cada test (*Delete*, *Update* y *Insert*).
 - Es importante puntualizar que este tiempo incluía el tiempo utilizado en el análisis del esquema y el requerido para completar las tres preguntas sobre el mismo.

Ejemplo

Planificación del Experimento



Ejemplo

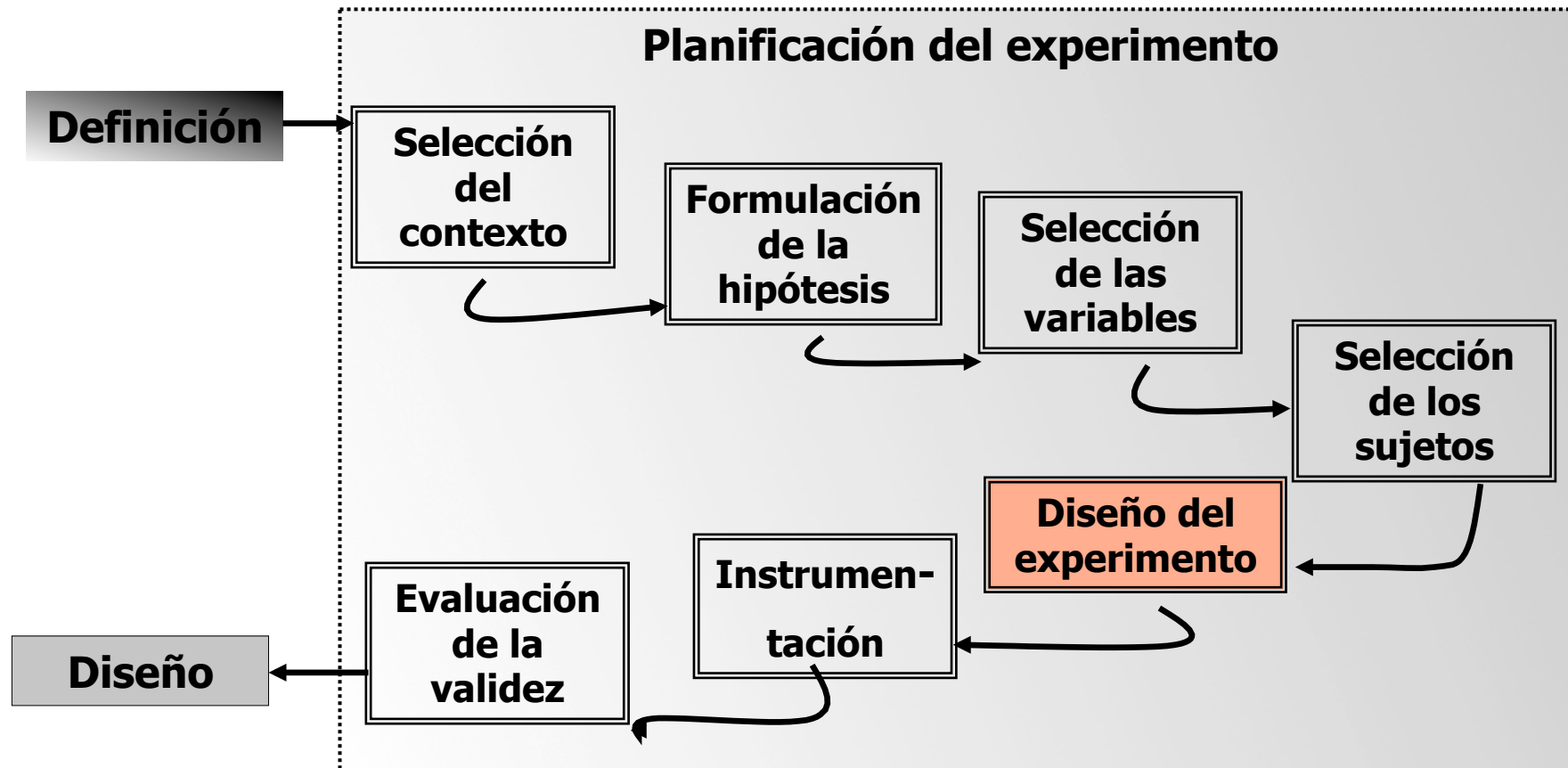
Planificación del Experimento

Selección de los sujetos

- 11 **profesionales** de una empresa realizaron el experimento
- Todos ellos tenían una **experiencia** media de 3 años trabajando con BD relacionales.

Ejemplo

Planificación del Experimento



Ejemplo

Planificación del Experimento

Diseño del experimento

- El experimento trabaja con **dos factores**: DRT y NFK, por lo que tenemos un modelo factorial.
- Ambas métricas pueden tomar **dos valores (tratamientos)**, 2 y 5 para la métrica DRT y 5 y 8 para la métrica NFK.

Diseño factorial

2 x 2

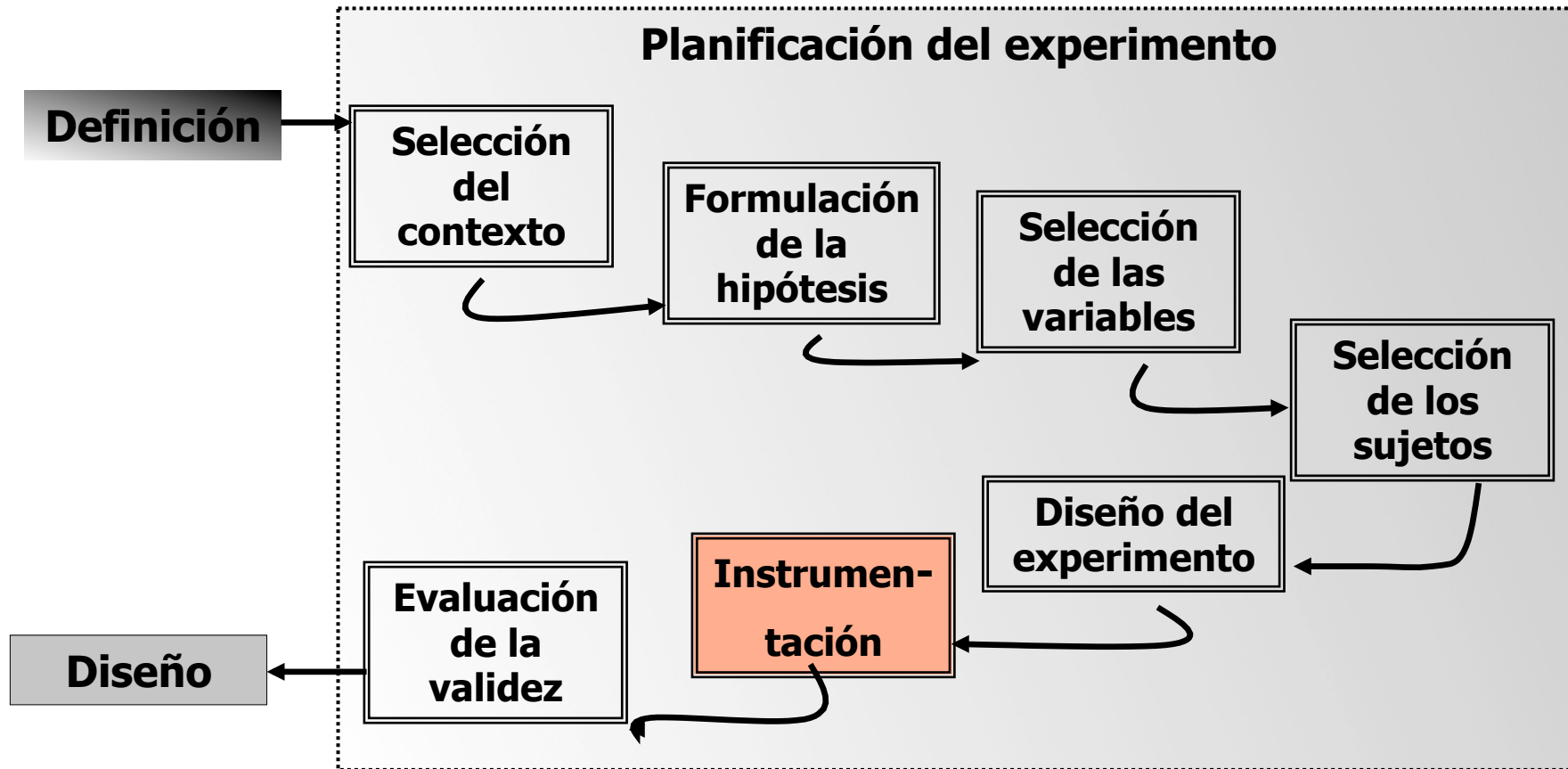
Intra-sujetos



UNIVERSIDAD
POLITECNICA
DE VALENCIA

Ejemplo

Planificación del Experimento



Ejemplo

Planificación del Experimento

Instrumentación

- Para el experimento se utilizaron **cuatro** esquemas de bases de datos relacional.
- Para cada diseño la documentación constaba de unas **7 páginas** que incluían además de los **esquemas** de las bases de datos, una **descripción general** y un **documento de requisitos**.
- Para conseguir que los **diseños** fueran comparables, tratamos de que fueran lo más **similares** posible.
- Para cada diseño había que **realizar 3 operaciones** (*Insert*, *Delete* y *Update*) y los sujetos debían anotar cuantas columnas de cada tabla iban a resultar afectadas como resultado de cada una de estas operaciones.



Ejemplo

Planificación del Experimento

Instrumentación

- Con los resultados obtenidos debían rellenar un cuestionario

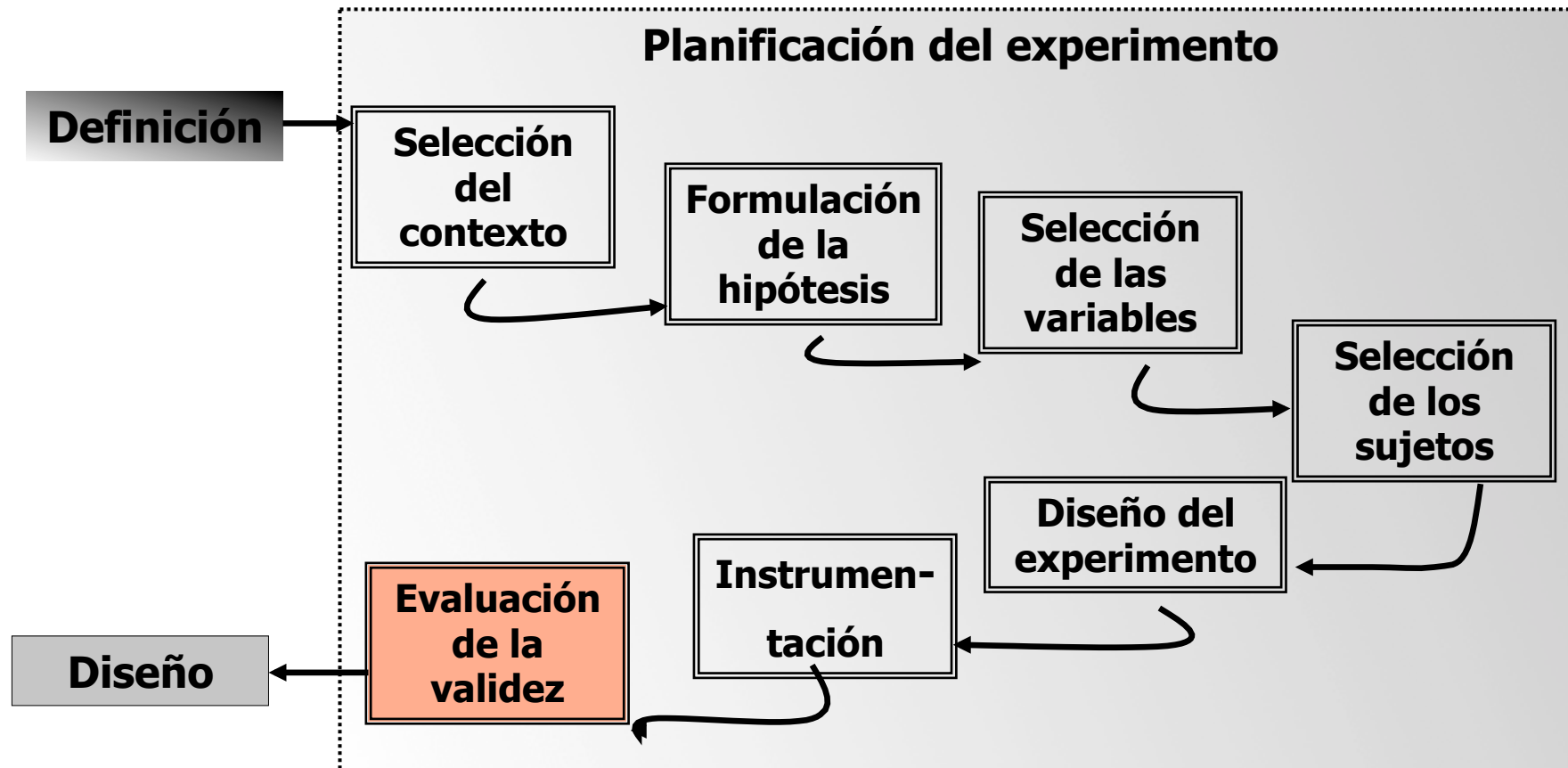
START TIME:					
1. What tables and how many rows in each table are affected if we delete in the Table 5 the row with code1=210?					
Table 1	Table 2	Table 3	Table 4	Table 5	Table 6
2. What tables and how many rows in each table are affected if we update the column X of the row with code2=11 in the table 3?					
Table 1	Table 2	Table 3	Table 4	Table 5	Table 6
3. What tables and how many rows and columns are necessary to add if we want add a new row in the table 4? (Suppose that all the necessary data are news in the database)					
Table 1	Table 2	Table 3	Table 4	Table 5	Table 6
END TIME:					



UNIVERSIDAD
POLITECNICA
DE VALENCIA

Ejemplo

Planificación del Experimento



Ejemplo

Planificación del Experimento

Evaluación de la validez

Validez de constructo

- Proponemos el **tiempo** para determinar el estado final de la base de datos (tras realizar las tareas) como medida razonable de la analizabilidad.
- Para asegurar la validez de constructo sería necesario realizar **más experimentos**, variando las operaciones a desarrollar.

Ejemplo

Planificación del Experimento

Evaluación de la validez

Validez Interna

- **Diferencias entre sujetos.** Los experimentos intra-sujetos reducen la variabilidad entre sujetos. En los experimentos todos los sujetos tenían, aproximadamente, la misma experiencia trabajando con bases de datos relacionales.
- **Diferencias entre esquemas.** Todos los esquemas fueron diseñados con seis tablas relacionadas con más o menos claves ajenas dependiendo de los valores de las métricas. Sin embargo, el dominio de los esquemas eran diferentes lo que pudo influenciar, de alguna forma, en los resultados obtenidos.



Ejemplo

Planificación del Experimento

Evaluación de la validez

- **Precisión de los valores del tiempo.** Los sujetos fueron los encargados de apuntar los tiempos de comienzo y fin de cada test. Pensamos que este método es más efectivo que tener un supervisor encargado de anotar los tiempos de cada sujeto. Sin embargo, somos conscientes de que el sujeto puede introducir alguna imprecisión.
- **Efectos de aprendizaje.** Los tests fueron colocados en distinto orden para los diferentes sujetos. De esta forma, cada sujeto debía contestar los tests en el orden que se le entregaban.
- **Efectos de fatiga.** El tiempo medio para completar el experimento fue una hora por lo que los efectos de fatiga son prácticamente inexistentes. Además, el orden diferente de los tests ayudaron a evitar este tipo de efectos.



Ejemplo

Planificación del Experimento

Evaluación de la validez

- **Efectos de persistencia.** Este tipo de efectos no se dieron ya que era la primera vez que los sujetos realizaban un experimento de estas características.
- **Motivación de los sujetos.** Se les indicó a los profesionales que los tests eran anónimos y que en ningún caso la información obtenida sería comentada entre los trabajadores ni entre los jefes.
- **Otros factores.** El plagio y la influencia entre los sujetos fueron controlados. Los sujetos fueron informados sobre la imposibilidad de hablar con otros sujetos y compartir respuestas. Sin embargo, no hubo ningún tipo de vigilancia por lo que no podemos asegurar que no aparecieran efectos del tipo comentado, aunque no parece posible.

Ejemplo

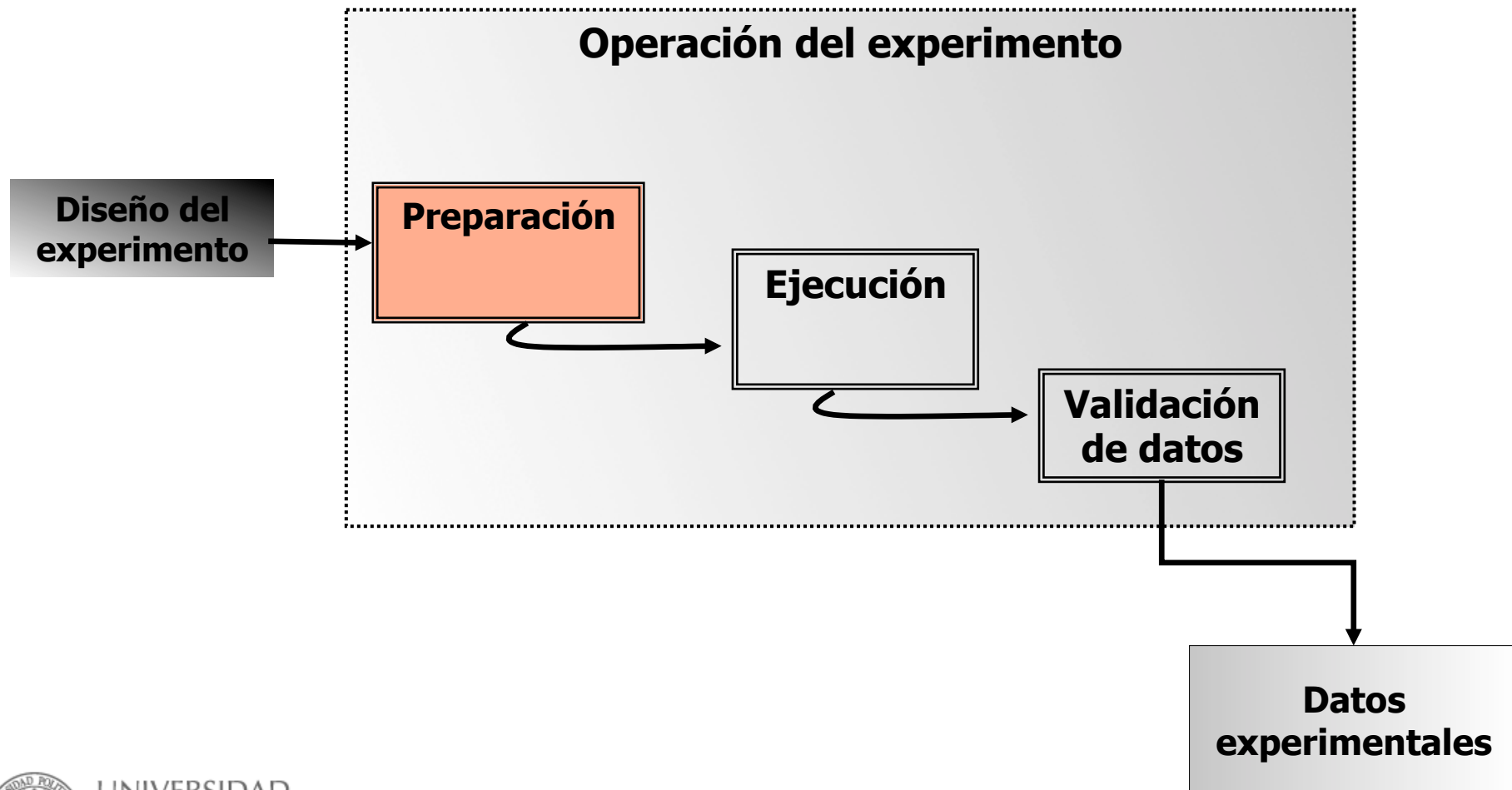
Planificación del Experimento

Evaluación de la validez

Validez externa

- **Materiales y tareas utilizados**
 - Se ha utilizado en el experimento esquemas y operaciones que sean representativos de casos reales.
 - Son necesarios más experimentos con BD relacionales de mayor tamaño y complejidad.
- **Sujetos**
 - Son necesarios nuevos experimentos con un mayor número de sujetos (estudiantes y profesionales) y con una mayor diferencia entre los valores de las métricas.

Ejemplo Operación

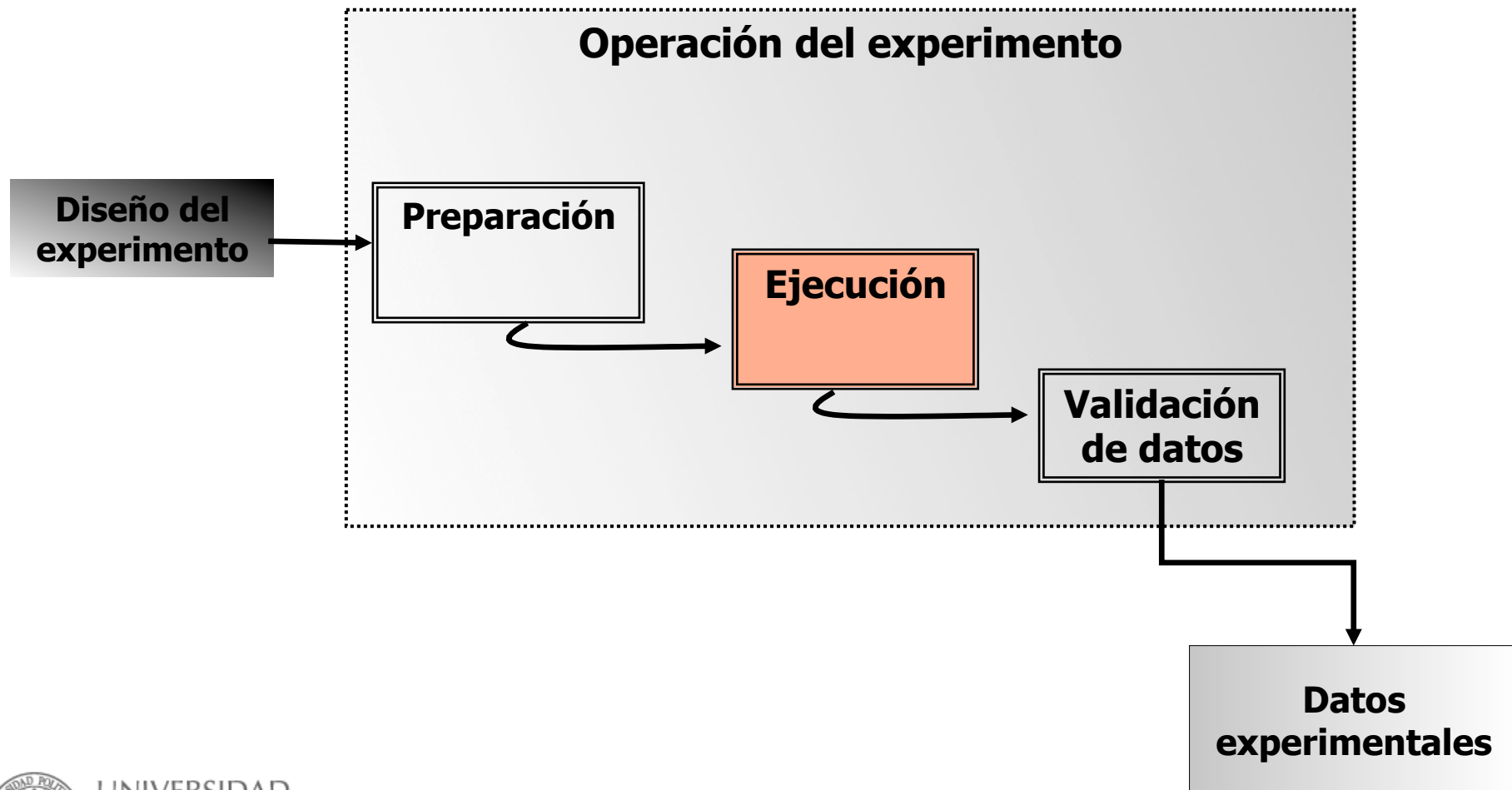


Ejemplo Operación

Preparación

- Antes de que los sujetos realizaran el experimento, este fue realizado por un **grupo reducido** para mejorarlo y asegurar que tanto el experimento como la documentación estaban bien diseñados.
- Como **resultado** de este pre-ejecución el único cambio necesario fue en el cuestionario que fue cambiado para hacerlo más fácil de utilizar (transformándolo en tabla).

Ejemplo Operación



Ejemplo

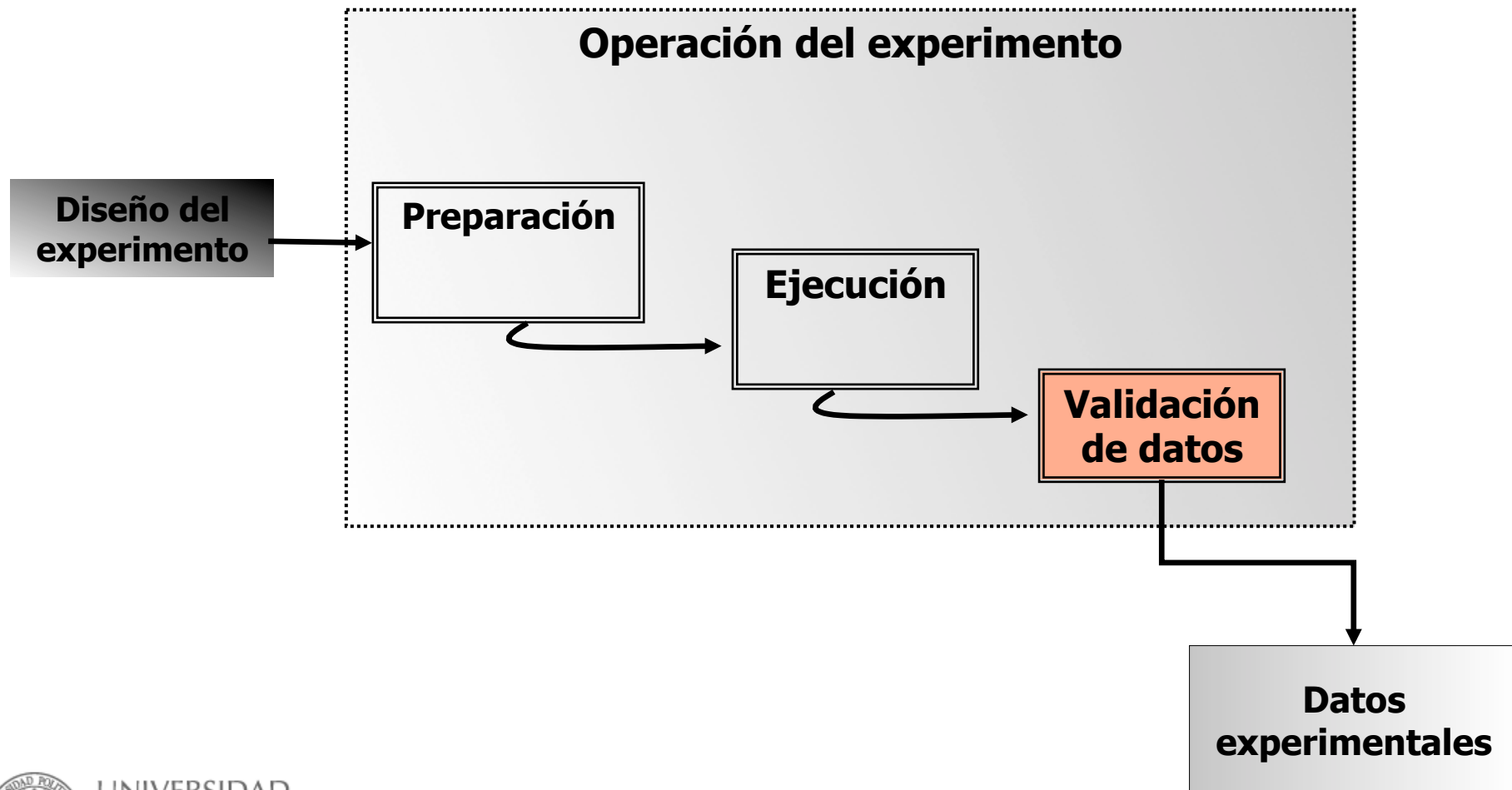
Operación

Ejecución

- Los tests se realizaron a lo largo de **una hora**. Se le explicó a los sujetos que tipo de ejercicios realizarían, el material que podrían usar, el tipo de respuestas que debían proporcionar y como debían anotar el tiempo utilizado en resolver los ejercicios.
- Antes de cada test, los sujetos debían anotar el **tiempo de comienzo** y, al terminar los ejercicios, el **tiempo de fin**.
- Los tests fueron realizados en **diferente orden** por los distintos sujetos para evitar los efectos de aprendizaje.



Ejemplo Operación



Ejemplo Operación

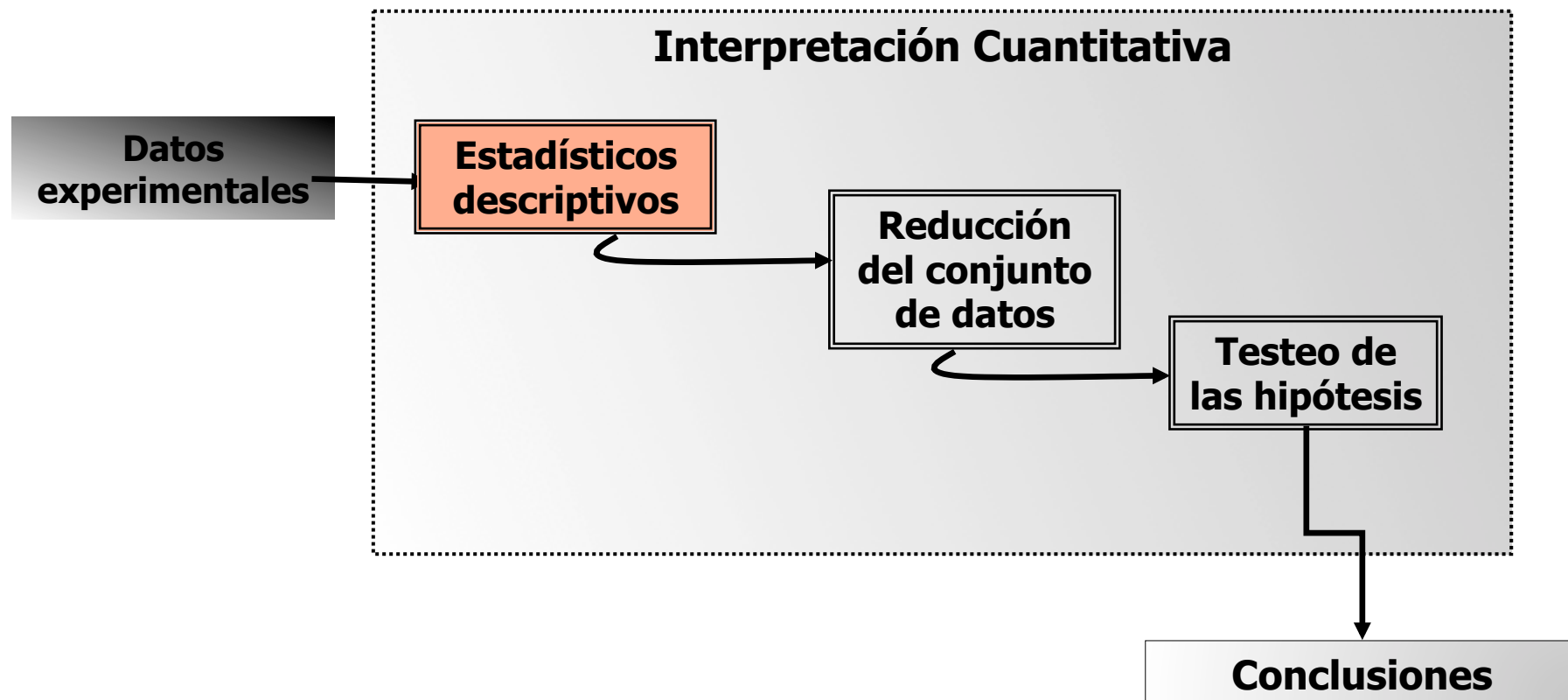
Validación de datos

- Como todos los tests fueron **contestados correctamente** pudimos utilizar el número de minutos utilizados por cada sujeto para la obtención de los resultados del experimento.
- Asimismo, por ser correctos todos los tests, contamos con **11 conjuntos de datos** para cada test, cada uno de los cuales corresponde a la respuesta de cada sujeto.



Ejemplo

Interpretación Cuantitativa



Ejemplo

Interpretación Cuantitativa

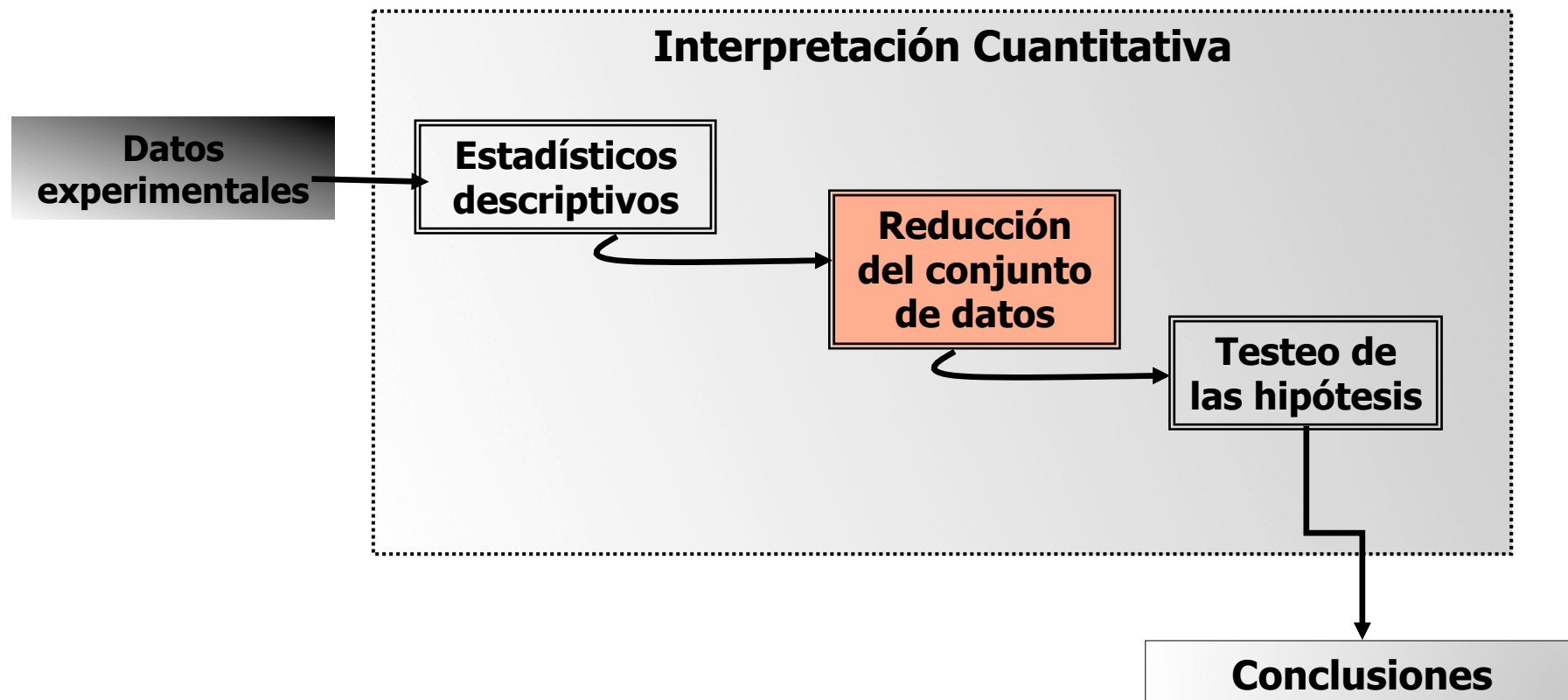
Estadísticos Descriptivos

	Mínimo	Máximo	Media	Desviación estándar
Caso Uno	4	16	7,1818	3,6005
Caso Dos	3	8	5,4545	1,8635
Caso Tres	6	15	10,0909	2,7002
Caso Cuatro	3	15	7,0909	3,7803



Ejemplo

Interpretación Cuantitativa

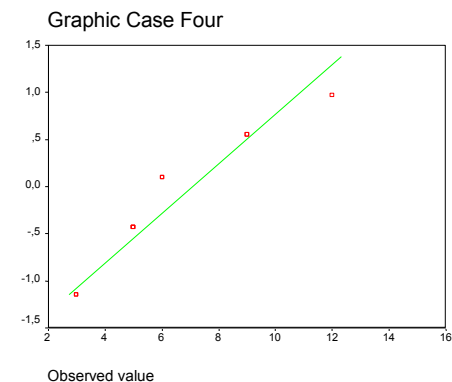
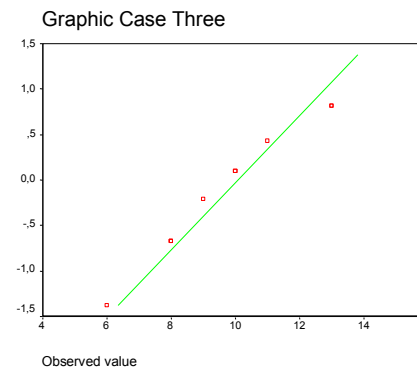
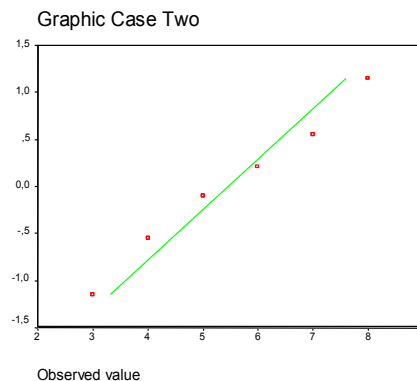
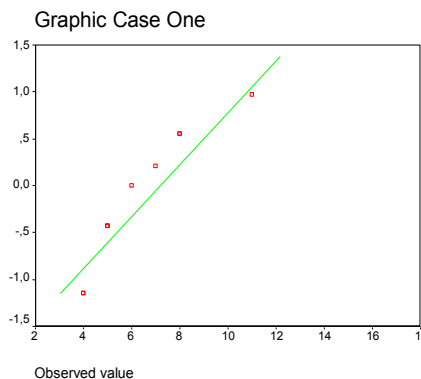


Ejemplo

Interpretación Cuantitativa

Reducción del conjunto de datos

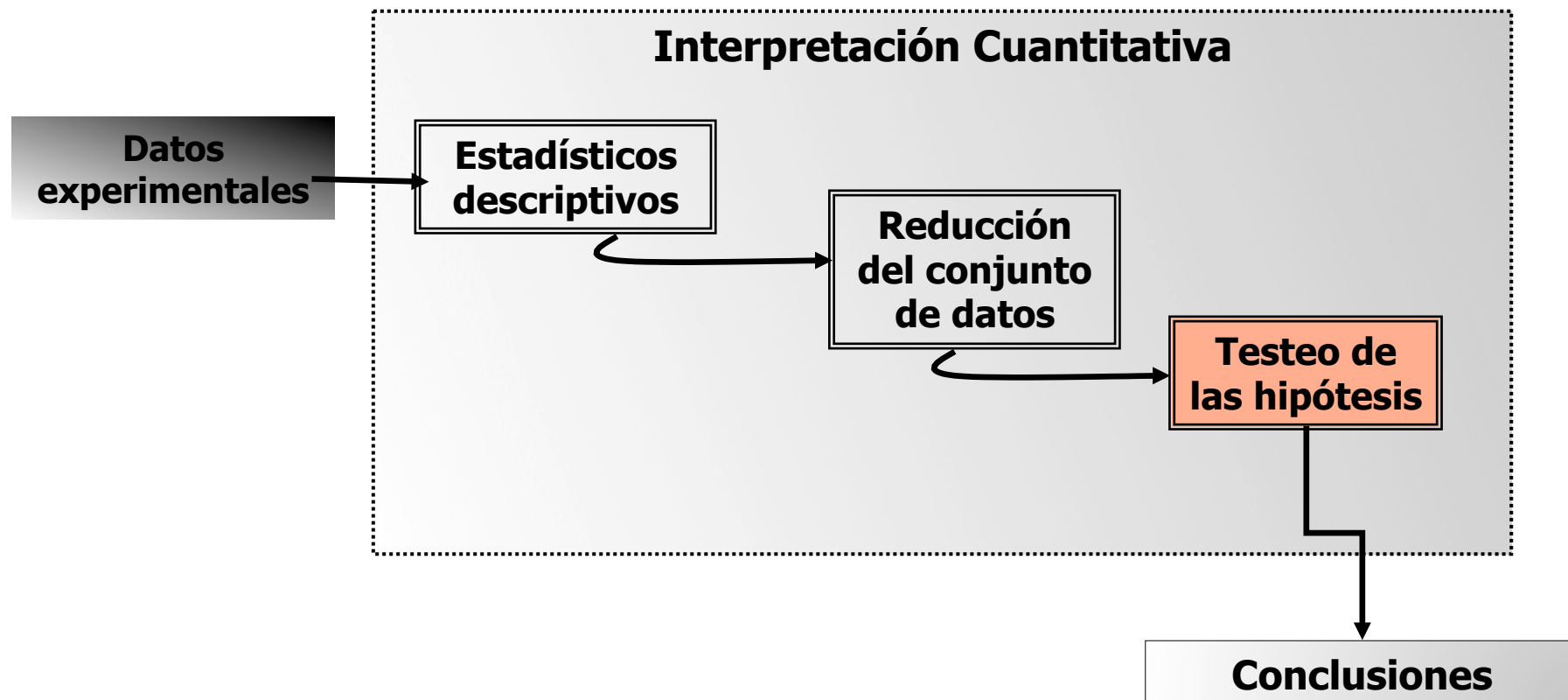
- Mediante el gráfico del test de normalidad, los puntos están cerca de la línea por lo que los datos tienen una **distribución normal** y, además, no hay *outliers*.



- También aplicamos los test de **Shapiro-Wilk** y de **Kolmogorov-Smirnov** encontrando también que los datos eran normales.

Ejemplo

Interpretación Cuantitativa



Ejemplo

Interpretación Cuantitativa

Prueba de las hipótesis

- Como los datos siguen una distribución normal, podemos utilizar **tests paramétricos** (que requieren menor cantidad de datos).
- Antes de proceder al análisis, es necesario fijar el **nivel de significación**. Se ha decidido establecer un valor de $\alpha = 0.1$ que significa un nivel del 90% de confianza.
- Después de todas estas consideraciones es posible seleccionar el mejor test para analizar los datos. Teniendo en cuenta las hipótesis, se ha seleccionado el test ANOVA.

Ejemplo

Interpretación Cuantitativa

Prueba de las hipótesis

Fuente de variación	Grados de libertad	F-Ratio	Sig.
DRT	1	9.137	0.013
NFK	1	12.659	0.005
Interacción	1	0.631	0.445
Error	1		
Total	40		

H1. El valor de la métrica DRT afecta a la analizabilidad del esquema de base de datos.

Como $9.137 > 2.84$, la hipótesis alternativa 1 es válida porque el valor de la métrica DRT influye en los resultados obtenidos.



Ejemplo

Interpretación Cuantitativa

Prueba de las hipótesis

Fuente de variación	Grados de libertad	F-Ratio	Sig.
DRT	1	9.137	0.013
NFK	1	12.659	0.005
Interacción	1	0.631	0.445
Error	1		
Total	40		

H2. El valor de la métrica NFK afecta a la analizabilidad del esquema de base de datos.

Como $12.659 > 2.84$, la hipótesis alternativa 2 es válida ya que el valor de la métrica NFK influye en los resultados obtenidos.



UNIVERSIDAD
POLITECNICA
DE VALENCIA

Ejemplo

Interpretación Cuantitativa

Prueba de las hipótesis

Fuente de variación	Grados de libertad	F-Ratio	Sig.
DRT	1	9.137	0.013
NFK	1	12.659	0.005
Interacción	1	0.631	0.445
Error	1		
Total	40		

H3. La interacción de las métricas DRT y NFK afecta a la analizabilidad del esquema de base de datos.

Como $0.631 < 2.84$, la hipótesis alternativa 3 es inválida ya que los valores de las métricas DRT y NFK no infuyen en los resultados obtenidos.

Ejemplo

Interpretación Cuantitativa

Prueba de las hipótesis

- Ambas métricas parecen ser **buenos indicadores de la analizabilidad** de esquemas de bases de datos relacionales.
- También se ha demostrado que **no existe interacción** entre ambas métricas.

Ejemplo

Interpretación Cuantitativa

Prueba de las hipótesis

- Como resultado del proceso experimental podemos concluir:
 - **NFK** parece ser un **buen indicador** de la analizabilidad de bases de datos relacionales.
 - Sin embargo, es **difícil** extraer una conclusión sobre **DRT**
- Sería imprescindible realizar **más experimentos** con las métricas para llegar a conclusiones más sólidas.

Conclusiones

- Sin realizar mediciones no existe una forma de determinar si el proceso/producto está mejorando.
- Las actividades de medición deben ser dirigidas por un objetivo.
- Las métricas permiten establecer objetivos significativos para la mejora.
- Las métricas permiten identificar las causas de defectos que tienen un mayor impacto en el desarrollo del software.
- Cuando las métricas son aplicadas a los productos software estás permiten identificar:
 - Que requisitos de usuarios pueden cambiar
 - Que módulos son los más propensos a errores
 - La cantidad de pruebas que deben ser planificadas para cada módulo.Etc.