

*El II Congreso Mundial Vasco, patrocinado por el Gobierno Vasco y organizado con la colaboración de las más prestigiosas instituciones académicas y culturales del País Vasco, se ha celebrado entre los meses de agosto y diciembre de 1987. Se trata de un encuentro multidisciplinar que ha agrupado a 34 Congresos y «Workshops», cubriendo diferentes áreas y campos de conocimiento.*

*El II Congreso Mundial Vasco ha contemplado un doble objetivo. En primer lugar, tratar temas de actualidad desde una perspectiva científica y universal, en el intento de encontrar soluciones a problemas de interés social, y en segundo lugar, ofrecer un marco adecuado para el fortalecimiento de vínculos entre el País Vasco y la Comunidad cultural y científica internacional, al mismo tiempo que ha servido de oportuna plataforma para el intercambio de ideas, métodos y recomendaciones en el campo de la Ciencia, la Técnica, la Cultura, etc.*

*En este marco, el Congreso de Educación, celebrado en Bilbao, entre el 13 y el 17 de octubre de 1987, ha reunido las más modernas aportaciones de los más prestigiosos especialistas en seis áreas que comprenden la temática educativa de nuestro momento: planificación de la educación y mercado de trabajo, la gestión educativa ante la innovación y el cambio, temas actuales sobre psicopedagogía y didáctica, perspectivas y problemas de la función docente, tecnología y educación, y aspectos metodológicos de la investigación educativa.*

Congreso de Educación / 6

Aspectos metodológicos de la investigación educativa

# Aspectos metodológicos de la investigación educativa

**IÑAKI DENDALUCE (Coord.)**

**Margarita Bartolomé  
Thomas Cook  
Törsten Husén  
Robert Linn  
Joan Mateo  
Mario de Miguel  
Arturo de la Orden  
Laura Peracchio  
Philippe Pilibossian  
Javier Tejedor  
.....**

M.I.D.E.  
A/120



II CONGRESO MUNDIAL VASCO



6



II CONGRESO MUNDIAL VASCO



## Una aplicación del Diseño de Facetas de Guttman como elemento en la construcción de un test asistido por ordenador (TAO-CAT)

J. M. Jornet y J. M. Suárez

La Teoría Estructural de Facetas (o Diseño de Facetas) fue formulada por Guttman (1959) en el ámbito de la Teoría de la Personalidad y aplicada posteriormente por este mismo autor (1965, 1969) a los Tests de Rendimiento. El Diseño de Facetas aparece sobre la base de que «... un constructo puede explicarse en términos de medida sobre una base ordenada y precisa...»; su esencia es que «... cualquier constructo puede describirse en términos de sus componentes» (Roid y Haladyna, 1982, p. 127). Con el Diseño de Facetas

«... primero, se especifica un sistema definicional para el Universo de Contenido y observaciones en la forma de una Sentencia Directriz (*Mapping Sentence*); segundo, se realizan las especificaciones acerca de las Facetas de la Sentencia Directriz...; las definiciones y especificaciones conllevan una hipótesis estructural que se comprueba mediante datos empíricos» (Guttman, 1969, p. 56).

Dentro de un Diseño de Facetas pueden identificarse los siguientes elementos: Componentes del Diseño y Niveles funcionales de desarrollo.

### COMPONENTES DE UN DISEÑO DE FACETAS

Sentencias Directrices, Facetas y Elementos. (Véase tabla 1.) Las Sentencias Directrices son expresiones resumidas que marcan la estructura (orden, componentes y subcomponentes) y límites de un constructo o Dominio Educativo (que en el caso de los Tests de Rendimiento puede ser un Objetivo Operativo). Dichas Sentencias incluyen partes fijas —similares a los *Item Form*— y variables —Facetas—, las cuales son series de reemplazamiento compuestas por diferentes variaciones de su contenido.

En el nivel superior de generalización, el conjunto de todas las Sentencias Directrices componentes de un Diseño de Facetas constituyen el Dominio o Universo de Medida especificado. En el nivel inferior, cada una de las posibles combinaciones elemento a elemento de cada Faceta de una Sentencia Directriz es un posible ítem. (Véase tabla 2.)

TABLA 1

### EJEMPLO DE SENTENCIA DIRECTRIZ Y SU DESARROLLO PARA LA GENERACION AUTOMATICA DE ITEMS

OBJETIVO: Dadas las órdenes elementales de disco del lenguaje BASIC (ej. LOAD, SAVE...) el alumno identificará su uso.

SENTENCIA DIRECTRIZ: «La orden (Faceta A) sirve para (Faceta B)»

FACETA A. Tiene 3 elementos:

1. LOAD «nombre del programa»
2. SAVE «nombre del programa»
3. KILL «nombre del programa»

FACETA B. Tiene 3 elementos:

1. Llamar a un programa que está en el disco.
2. Grabar un programa en el disco.
3. Borrar un programa del disco.

ALGORITMOS DE GENERACIÓN. 1. Aleatorizar si se incluye como V o F (1-2). 2. Si se obtiene en (1) 1, entonces: Se aleatorizan los elementos de A correspondientes al valor correcto de B. 3. Si se obtiene en (1) 2, entonces: Se aleatorizan los elementos de A correspondientes a los no extraídos de B.  
Nº DE ITEMS POSIBLE:  $3 \times 3 = 9$  TIPO DE ITEMS: Verdadero-Falso

MODULO ALGORITMICO DE GENERACION POR COMPUTADOR. El presente módulo genera aleatoriamente los ítems pertenecientes a este objetivo según las condiciones expuestas. Igualmente, ofrece al evaluador la solución de la afirmación (V/F).

```

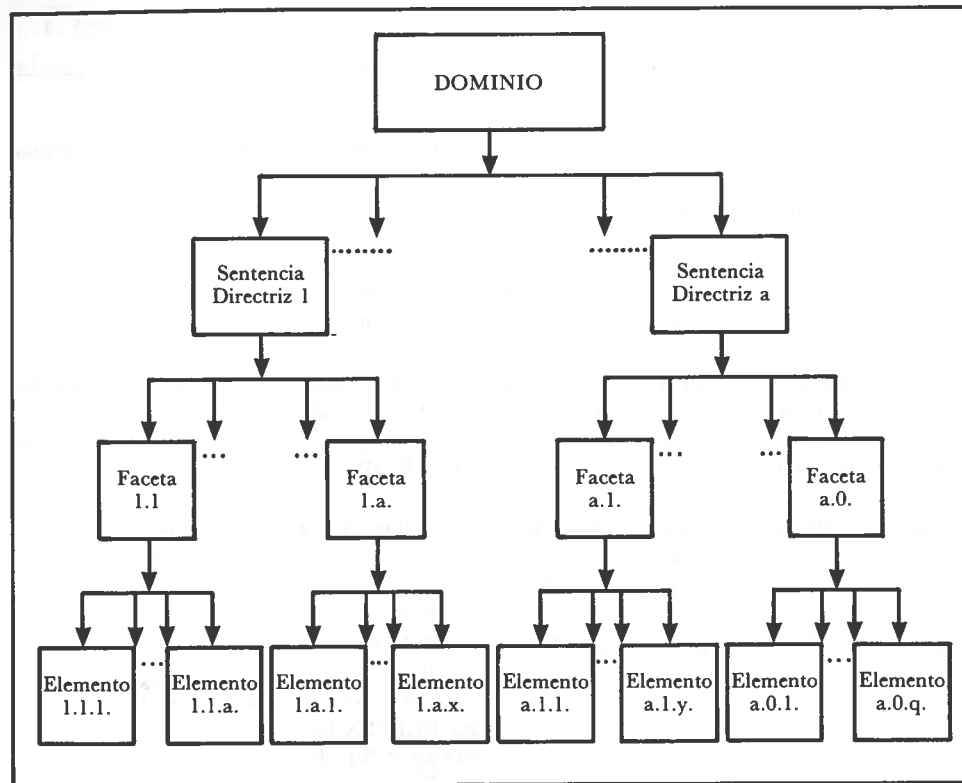
10 REM AS=Vector Faceta A,BS=Vector Faceta B,CS=Vector V/F      250 RETURN
20 J=RND (1): IF J>.5 THEN J=1: GOTO 70                          260 PRINT «La orden»; AS (L); «sirve para»; BS
    ELSE J=0                                                       (K); «...V F»
40 GOSUB 110                                                       270 PRINT «La cuestión es»; CS(J+1)
50 L=K                                                             280 RETURN 100
60 GOSUB 260                                                       290 X1=2:X2=3
70 GOSUB 110                                                       300 GOSUB 230
80 ON K GOSUB 290, 320, 360                                       310 RETURN
90 GOSUB 260                                                       320 X1=1:X2=3
100 END                                                            330 GOSUB 230
110 K=1+RND(1)*2                                                  340 IF L=2 THEN 320
120 IF K-INT (K)>.5 THEN K=INT(K)+1                               350 RETURN
    ELSE K=INT (K)
130 RETURN                                                         360 X1=1:X2= 2
230 L=X1+(X2 -X1)*RND(1)                                          370 GOSUB 230
240 IF L-INT (L)>.5 THEN L=INT (L)+1                              390 RETURN
    ELSE L=INT(L)

```

EJEMPLO DE ITEM: A2 B2: La orden SAVE «Nombre del programa» sirve para grabar un programa en el disco... V F (Verdadero)

TABLA 2

DESCOMPOSICIÓN INTERNA SIMPLE DE UN DOMINIO A PARTIR DE UN DISEÑO DE FACETAS



### NIVELES FUNCIONALES DE DESARROLLO

Para este cometido existen algunas revisiones y consejos de gran utilidad: Engel y Martuza (1976); Runkel y McGrath (1972); Berk (1978); Roid y Haladyna (1982). En todos ellos aparecen diversas recomendaciones respecto a la aplicación del Diseño de Facetas como un análisis de contenido que permite generar *ítems* los cuales teóricamente presentan *a priori* los mismos niveles de dificultad y discriminación. Lo que no es totalmente cierto y depende de la calidad y extremado cuidado en la definición de la Sentencia Directriz (Jornet, 1987).

Entre ellos, quizá Roid y Haladyna (1982) son los que resumen más nítidamente la secuencia ideal de desarrollo de un Diseño de Facetas, la cual sintetizamos a continuación:

1. *Selección de un objetivo de Instrucción:* Engel y Martuza (1976) y Berk (1978) sugieren que las Sentencias Directrices se pueden desarrollar a partir de Objetivos de Instrucción.

2. *Listado del Material de Instrucción:* Para desarrollar el Diseño de Facetas Roid y Haladyna (1982) aconsejan que se disponga del material de instrucción, dado que supone comúnmente la descripción detallada del propósito de la misma.
3. *Desarrollar un Objetivo Amplificado:* Al estilo de Popham (1978), dado que provee una descripción general del contenido a especificar.
4. *Generar la Sentencia Directriz:* A partir del Objetivo Amplificado se pueden identificar los componentes del contenido que pueden incluir variaciones (Facetas) y se construye la Sentencia Directriz.
5. *Generar la Estructura de Facetas del ítem:* En este punto es preciso tener en cuenta que tanto el número de elementos de cada Faceta, como el número de estas condicionan el número de *ítems* resultantes. Así, a mayor número de elementos y facetas, mayores posibilidades de combinación y, por lo tanto, mayor número de *ítems*.
6. *Escribir el ítem:* Este será cualquier combinación de elementos de cada Faceta en la Sentencia Directriz. Runkel y McGrath (1972, p. 12) presentan las reglas para la escritura de *ítems* con el Diseño de Facetas, que sintetizamos a continuación:

- a) Todas las propiedades o facetas que se han elegido para un concepto deben aplicarse a todos los objetos clasificados.
- b) Cada faceta debe dividirse en todos sus posibles elementos.
- c) Los elementos deben ser mutuamente exclusivos y no deben solaparse.
- d) Las relaciones entre las facetas deben ser lógicas y deben estar bien descritas.
- e) Las relaciones entre los elementos también deben ser lógicas y deben estar bien descritas.
- f) Las Facetas deben cubrir exhaustivamente la definición del Dominio y representar la esencia del mismo, tal cual se concibe dentro del proceso de instrucción.

### CRITERIOS DE GENERACION AUTOMATICA

Una vez formuladas las Sentencias Directrices —incluyendo Facetas y Elementos— es posible establecer los criterios de Generación Automática y, posteriormente, escribir los Programas de Ordenador. Sin embargo, antes de especificar tales criterios, queremos señalar que, en base a nuestra experiencia estimamos que es posible distinguir dos tipos de Facetas: Exclusivas y Combinadas. Esta distinción se relaciona con la forma en que su combinación elicitá distractores.

- Así, una Faceta es Exclusiva cuando se refiere a un concepto en cuya definición pueden asumirse diversas características verdaderas, de forma que los distractores se deben generar como contraejemplos de las definiciones. (Véase tabla 3). Normalmente son simples, corresponden a definiciones de contenidos y generan elementos de Verdadero/Falso.
- Nos referimos a Facetas Combinadas, en relación a aquellas que, incluyendo diversas variaciones que corresponden a elementos variables de otra unidad de contenido o Faceta, los distractores se generan a partir de las combinaciones de los elementos no-correspondientes de dos o más Facetas. (Véase tabla 3). Suelen ser complejas, corresponden a asociaciones de contenidos y pueden generar elementos de cualquier tipo (V/F, Elección Múltiple o Producción).
- Junto a estos tipos, los requerimientos de la generación de distractores pueden conllevar el uso conjunto de Facetas Exclusivas con Combinadas. (Véase tabla 3).



TABLA 3

EJEMPLOS DE TIPOS DE FACETAS Y SENTENCIAS DIRECTRICES

CASO 1. Sentencia Directriz con Faceta Exclusiva. EJEMPLO:

OBJETIVO: El alumno reconocerá qué es la C.P.U.

SENTENCIA DIRECTRIZ: «La Unidad Central de Proceso (C.P.U.) es Faceta A (A<sub>1</sub> o A<sub>2</sub>)»

FACETA A<sub>1</sub>: Características de la definición de C.P.U. FACETA A<sub>2</sub>: Distractores de la definición:

- |  |  |
|--|--|
| 1. Una parte del Ordenador                     | 1. Todo el Ordenador                         |
| 2. Como el cerebro del Ordenador               | 2. Como las piernas que mueven el Ordenador  |
| 3. Una pastilla pequeña (1.5 cm <sup>2</sup> ) | 3. Una pastilla grande (10 cm <sup>2</sup> ) |
| 4. De Silicio                                  | 4. De diamante                               |

CASO 2. Sentencia Directriz con Facetas Combinadas. EJEMPLO 1: Véase tabla 1. EJEMPLO 2:

OBJETIVO: Dadas las asignaciones a tres variables numéricas en forma de un programa de cinco pasos el alumno reconocerá cuál será el estado final de las mismas tras su ejecución por el ordenador

SENTENCIA DIRECTRIZ: «Dadas las siguientes asignaciones a las variables, cuyos valores iniciales son Faceta D.

- 5 asignaciones:
- 1) Faceta A Faceta B Faceta C (Faceta B)
  - 2) Faceta A Faceta B Faceta C (Faceta B)
  - 3) Faceta A Faceta B Faceta C (Faceta B)
  - 4) Faceta A Faceta B Faceta C (Faceta B)
  - 5) Faceta A Faceta B Faceta C (Faceta B)

señalar de entre las siguientes afirmaciones cuál refleja el estado final de las variables después de su ejecución por el ordenador»

OPCIONES:

- |    |                               |   |   |
|----|-------------------------------|---|---|
|    | A                             | B | C |
| a) |                               |   |   |
| b) | Según algoritmo de generación |   |   |
| c) |                               |   |   |
| d) | Ninguna de las anteriores     |   |   |

FACETA A: Variables	FACETA B: Valores	FACETA C: Operadores aritméticos	FACETA D: Valores iniciales
1. A=	1. 0    2. 1	1.	1. A=4, B=6 y C=5
2. B=	3. 2    4. 3	2. +	2. Todos cero
3. C=	5. A    6. b	3. -	
	7. C	4. *	

supondría un aumento notable en la dificultad de los ítems.

ALGORITMOS DE GENERACION: 1. Asignaciones: 1.1. Se generan aleatoriamente con reemplazamiento los elementos de las Facetas A, B y C. 2.2. Se genera el elemento de la Faceta D condicionado a: 2.2.1. En el caso en que dos de las tres primeras asignaciones contengan ceros en las variables, se otorga elemento 1. 2.2.2. En el caso en que 2.1. no se dé, se otorga elemento 2. 2. Opciones: a, b y c. Se extraen aleatoriamente y sin reemplazamiento de los siguientes parámetros:

	A	B	C	A	B	C
V=Valor correcto	1. V	V	V	5. V	V-1	V
	2. V+1	V	V	6. V	V	V+1
	3. V-1	V	V	7. V	V	V-1
	4. V	V+1	V			

d. Es constante

Si al extraer aleatoriamente el elemento de la Faceta C sale el 1, no se extrae por segunda vez un elemento de la Faceta B, ya que al no haber operador se trata de una asignación directa.

CASO 3. Sentencia Directriz con Facetas Exclusivas y Combinadas. EJEMPLO:

OBJETIVO: Dadas una variable, una condición y dos acciones, el alumno formulará en BASIC una transferencia condicional con las acciones asociadas al cumplimiento e incumplimiento de la condición.

SENTENCIA DIRECTRIZ: «Dada una variable Faceta A establece una acción condicionada de Faceta B siempre que esta variable sea Faceta C y otra de Faceta D cuando esto no se cumpla»

SOLUCION:

FACETA A: Variables	FACETA B: Acciones (I)	FACETA C: Operadores de relación	FACETA D: Acciones (II)
1. A	FACETA B <sub>1</sub> :	1. Igual a 8	Faceta D <sub>1</sub> :
2. AS	1. Salto a la línea 500	2. Distinto a 8	1. Salto a la línea 600
	2. Incrementar en una unidad la variable A	3. Mayor a 8	2. Decrementar en una unidad la variable A
	3. Añadir a la variable A, la variable B	4. Menor a 8	3. Sustraer a la variable A la variable B
	FACETA B <sub>2</sub> :	5. Menor o igual a 8	FACETA D <sub>2</sub> :
	1. Salto a la línea 500	6. Mayor o igual a 8	1. Salto a la línea 600
	2. Añadir a la variable AS la constante «MADRID»		2. Añadir a una constante el valor de la variable AS y guardarlo en la misma.
	3. Añadir a la variable AS la variable BS		3. Añadir a la variable AS la variable BS y guardarlo en BS

ALGORITMOS DE GENERACION: 1. Aleatorizar los elementos de la Faceta A (1-2). 2. Si en [1] se obtiene Faceta A=1, entonces: Faceta B=Faceta B<sub>1</sub>, y Faceta D=Faceta D<sub>1</sub>; Se extraen aleatoriamente los elementos de Faceta B<sub>1</sub> (1-3), Faceta C (1-6) y Faceta D<sub>1</sub> (1-3). 3. Si en [1] se obtiene Faceta A=2, entonces: Faceta B=Faceta B<sub>2</sub> y Faceta D=Faceta D<sub>2</sub>; Se extraen aleatoriamente los elementos de Faceta B<sub>2</sub> (1-3), Faceta C (1-2) y Faceta D<sub>2</sub> (1-3).

NOTA: Se aleatorizan únicamente las dos primeras relaciones condicionales, puesto que son las simples y usuales y, se corresponden con el nivel dado en el curso ya que para las otras relaciones se requiere un mayor nivel de profundización (valor ASCII y operaciones con strings)

EJEMPLOS DE ITEMS

CASO 1. La Unidad Central de Proceso (CPU) es de Silicio (V)

CASO 2. Dadas las siguientes asignaciones a las variables, cuyos valores iniciales son todos cero,

```
10 LET B=2
20 LET A=3
30 LET C=0
40 LET C=A*B
50 LET A=C-B
```

Señalar de entre las siguientes afirmaciones cuál refleja el estado final de las variables después de su ejecución por el ordenador:

- Opciones:
- |                              |   |   |                       |
|------------------------------|---|---|-----------------------|
| A                            | B | C |                       |
| -4                           | 2 | 6 | a (Solución correcta) |
| -4                           | 2 | 7 | b                     |
| -3                           | 2 | 6 | c                     |
| — Ningun de las anteriores d |   |   |                       |

CASO 3: Dada una variable A establece una acción condicionada de incrementar en una unidad la variable A, siempre que esta variable sea distinta a 8 y, otra de salto a la línea 600 cuando esto no se cumpla: SOLUCION: IF A<>8 THEN A=A+1 ELSE GOTO 600

Esta distinción es importante dado que está a la base de los criterios de Generación Automática. Así, considerando que los *ítems* corresponden a cualquier combinación de elementos de las Facetas es preciso tener en cuenta que la probabilidad de que se den combinaciones que correspondan a *ítems* cuya formulación sea verdadera o falsa no es siempre la misma. Atendiendo pues a las diversas notas expuestas, llegamos a una simplificación de criterios que estimamos satisface la diversidad de situaciones que pueden darse con el uso del Diseño de Facetas. La secuencia es la siguiente:

1. Aleatorizar si se incluye una opción cuya formulación sea:

- En el caso de *ítems* de V/F, Verdadera o Falsa.
- En el caso de *ítems* de Elección Múltiple, una opción verdadera, o asumir una última opción en la que se desestiman las anteriores.
- En el caso de *ítems* de Producción, una formulación que haga (o no) posible la producción.

2. Determinado el punto anterior, se determinan las asociaciones de elementos entre Facetas. Se entiende que en el caso de incluir una opción verdadera se aleatoriza dentro del marco restringido de asociaciones correctas y, viceversa para el caso contrario.

En relación a la escritura de Programas de Generación Automática es la propia estructura del Diseño de Facetas la que permite tal operativización (Millman, 1980, 1984; Roid, 1984). En nuestro caso, el ejemplo presentado en la tabla 1 está realizado en MBASIC sobre Sistema Operativo MS-DOS e incluye tanto el módulo de generación como una rutina de control, de forma que se ofrece al evaluador tanto la formulación del *ítem* como la respuesta correcta.

Este método permite, pues, construir para cada *ítem* un Banco de Generación Automática por Ordenador; de forma que convenientemente estructurados de acuerdo con el Diseño de Facetas pueden constituir los elementos básicos de construcción de un Test Administrado por Ordenador (TAO-CAT). Esta metodología ha sido probada con éxito (Jornet, 1987) en relación a un Curso de BASIC, como un instrumento de Generación de *Items*. Todo ello constituye, asimismo, un paso previo y un sistema que permite la determinación subsiguiente de las variables implicadas, sus relaciones estructurales y características métricas que posibilitan abordar satisfactoriamente una prueba del tipo CAT en sentido pleno.

#### REFERENCIAS BIBLIOGRAFICAS

- BERK, R. A. (1978): «The application of structural facet theory to achievement test construction». *Educational Research Quarterly*, 3, 62-72.
- ENGEL, J. D. y MARTUZA, V. R. (1976): *A systematic approach to the construction of Domain-Referenced Multiple-Choice test items*. Paper presented at the Meeting of the American Psychological Association, primavera, Washington.
- GUTTMAN, L. (1959): «A structural theory for intergroup beliefs and actions». *American Sociological Review*, 24, 318-328.
- (1965): *The structure of interrelations among intelligence tests*. Proceedings of the 1964. Invitational Conference on Testing Problems. Educational Testing Service, Princeton, Nueva Jersey.
- (1969): *Integration of test design and analysis*. Proceedings of the 1969 Invitational Conference on Testing Problems. Educational Testing Service. Princeton, Nueva Jersey.
- JORNET, J. M. (1987): *Una aproximación teórico-empírica a los métodos de medición de referencia criterial*. Tesis Doctoral no publicada, Universidad de Valencia.

- MILLMAN, J. (1980): «Computer-based item generation», en BERK, R. A. (ed.): *Criterion-referenced Measurement*. Johns Hopkins University Press, Baltimore, Maryland.
- ROID, G. H. (1984): «Generating the test items», en BERK, R. A. (ed.): *A guide to criterion-referenced test construction*. The Johns Hopkins University Press, Baltimore, Maryland.
- ROID, G. H. y HALADYNA, T. M. (1982): *A technology for test-item writing*. Academic Press, Nueva York.
- RUNKEL, P. J. y McGRATH, J. E. (1972): *Research on Human Behavior: a Systematic Guide to Method*. Holt, Nueva York.

