Nathan Stemmer

TEORIA DE LA DEFINICION EN SISTEMAS RUSSELLIANOS*

1. EL SISTEMA SUPERIOR DE FUNCIONES

1*a*. INTRODUCCIÓN.

ESTUDIAMOS EN este trabajo la teoría de la definición respecto a sistemas formales de tipo russelliano. Aunque no nos ceñiremos estrictamente a un sistema determinado, trataremos principalmente el sistema superior de funciones, cuyas características principales se indicarán en este capítulo. El segundo capítulo está dedicado principalmente a la determinación del criterio que aceptamos para considerar una expresión (o un conjunto de expresiones), como una definición. Las características más importantes de este criterio son que esta expresión debe cumplir con la condición de eliminabilidad y de conservatividad.

Después se estudian las definiciones de predicados, y se exponen las condiciones formales que éstas deben cumplir para estar de acuerdo con el criterio aceptado. Los meta-teoremas MT4-1 a MT4-5 demuestran que efectivamente las condiciones indicadas cumplen con el criterio (capítulos 3 y 4).

En seguida se estudian las diferentes condiciones formales para la definición de functores, y se tratan los problemas específicos que presentan éstos (capítulos 5 y 6).

En relación a la introducción de estos functores a partir de un esquema sentencial, estudiamos los métodos de Hilbert y Bernays, de Russell y de Frege para resolver las dificultades inherentes a esto (Cap. 7). Por último, se analizan las definiciones de símbolos individuales, de operadores y otros (Cap. 8).

Usaremos las siguientes abreviaciones:

D4-5	para	Definición Nº 5 del capítulo 4.
Ej.7-2	"	Ejemplo Nº 2 del Cap. 7.
Cond. . .		Condición . . .
M.T. . . .		Meta-teorema . . .
Crit. . . .		Criterio. . .
Proc. . . .		Procedimiento . . .

* Primera Parte del trabajo presentado por el autor para optar al título de Profesor Extraordinario de Lógica Simbólica de la Facultad de Ciencias Físicas y Matemáticas (N. de la R.).

1*b*. REGLAS DE FORMACIÓN PARA EXPRESIONES.

Tomamos como base para nuestro estudio el sistema superior de funciones, pero frecuentemente nos restringiremos al sistema funcional básico (o de primer orden). Consideramos estos dos sistemas, sin mencionarlos especialmente como aplicados, en el sentido que pueden presentarse símbolos de constantes, tanto proposicionales, como individuales, predicados y functores, con o sin axiomas adicionales. (En especial tenemos el sistema funcional básico ampliado:

a. con los axiomas de igualdad y el símbolo "=".

b. con los axiomas de la aritmética y los símbolos numéricos).

No haremos una exposición sistemática del sistema superior de funciones, sólo indicaremos algunas características principales de éste. Esto nos servirá también para aclarar la terminología que usaremos. (Aquellos términos que son de uso habitual en lógica simbólica no se indicarán en forma especial).

Símbolos (o signos). Los símbolos que figuran antes del ";" son los que usaremos principalmente, los que figuran después no se usarán en forma sistemática. Entre estos últimos tendremos frecuentemente signos compuestos por varias letras del alfabeto, que se considerarán por convención, como un solo signo. Como norma general tenemos: si el signo habitual es una letra mayúscula (minúscula), estos últimos *empezarán* con mayúscula (minúscula).

1. Símbolos proposicionales.
   a. variables ("p", "q", "r", ...)
   b. constantes ("p_a", .. "q_b", ...; "t", "f", ...)

2. Símbolos individuales.
   a. variables ("x", "y", "z", ..., "u", "v", ...)
   b. constantes ("a", "b", "c", ...; "*sócrates*", ..., "0", "1," "2",..)

3. Predicados n-posicionales (o símbolos de funciones proposicionales n-posicionales)
   a. variables ("F", "G", .. "H", .. "K", ..)
   b. constantes ("P", "Q", .. "R", ...; "*Par*", .., "*Suma*", ., "$>$", ..)

4. Functores.
   a. variables ("*f*", "*g*", "*h*", ...)
   b. constantes ("*fu*", . . ; "*h*", . . , "*padre*", . . , "+", "—", " : ", . . .)

5. Conectivas (constantes). ("∼", "v", "⊃", " . ", "≡", ; "/", . . .)
6. Operadores (constantes). ("*E*", "λ", "ι", . . .)
7. Símbolos especiales (const.) ("(", ")", ",")

8. Además consideramos como signos del sistema respectivo, los que se introducen por definición. Se deberá indicar en cada caso a qué categoría pertenecerá.

Los signos de las clases 2, 3 y 4 se llaman *símbolos* (variables o constantes) *nombrativos*. Estos se clasifican en tipos y órdenes según la teoría habitual de los tipos. En los predicados usaremos preferentemente "*F*", "*G*", "*P*" y "*Q*" como uno-posicionales, "*H*" y "*R*" como bi-posicionales y "*K*" como n-posicional de un orden mayor a 1.

Combinaciones de símbolos. Entre las diferentes secuencias de símbolos que podemos formar, nos interesan los siguientes:

1. operadores. (ej. "*(Ex)*", "*(x)*", "*(x,y)*", "*(λx,F,K)*"; "*(ιF)*")
2. argumentos. (ej. "*(y)*", "*(x,a,Par)*", "*(doble(x),5)*")
3. esquemas.

   A. esquemas nombrativos.
      a. los signos de las clases 2, 3, y 4.
      b. un functor seguido por su argumento. (ej. "*f(h,x)*",)
      c. un λ —operador seguido por su operand. (ej. "*(λx,)x—y*")
      d. un ι —operador seguido por su operand. (ej. "*(ιy) p*")
      (Este último a veces lo llamamos esquema nombrativo ambiguo (ver 8*c*)).

   B. esquemas sentenciales (o expresiones bien formadas).
      (ej. "*Fx*", "*x* + 5 = *a*", "*(F) K(F,a,fu)*").

      Si en un esq. nombrativo no figuran variables libres lo llamamos *nombre* (en sentido estricto).
      (ej. "*a*", "*socrates*", "*Par*", "*(λx) x = 5*", "*doble(6:3)*").
      Si en un esq. sentencial no figuran variables libres lo llamamos *frase*. (ej. "$p_a \supset q_a$", "*(K) (EG) K(G)*").

4. operands. (son esq. nombrativos o esq. sentenciales) .

Algunas observaciones:

1. El nombre "operador" es ambiguo, ya que se refiere tanto al signo como a la combinación formada por los paréntesis, el signo y las variables. Sin embargo, esto no producirá dificultades, ya que el contexto indicará a qué se refiere.

2. Los operands para los operadores son siempre esq. sentenciales excepto para el λ —operador, que puede ser tanto un esq. sentencial como un nombrativo. (En el primer caso, la λ —expresión, es decir el λ —operador con su operand, es un predicado, en el segundo caso es un functor) .

3. En vez de "esq. nombrativos" se habla a veces de "expresiones del sistema de tipos" (Carnap) . (En inglés frecuentemente se usa "terms") .

4. En lugar de "esq. sentenciales" (o "expresiones bien formadas") usaremos muchas veces "expresiones" o "fórmulas", en aquellos casos en que no se presentan dudas. Pero a veces servirán para indicar ciertas secuencias de símbolos que no forman esq. sentenciales.

5. Algunos sistemas disponen también de functores parciales y de esq. nombrativos ambiguos. (Ver Cap. 7) .

1*c*. META-LENGUAJE.

Como meta-lenguaje usaremos:

A. El Castellano suplementado con las siguientes expresiones:

$Esq, \ldots Esq_1, \ldots$	para esquemas (tanto sentenciales como nombrativos)
$Esn, \ldots Esn_1, \ldots t, \ldots t_1, \ldots s, \ldots$	para esquemas nombrativos.
$Ess, \ldots A, B, C, \ldots$	para esquemas sentenciales.
$v, \ldots v_1, \ldots w_n, \ldots u, \ldots$	para variables de cualquier tipo.
$V, \ldots V_1, \ldots W, \ldots$	para variables de cualquier tipo, pero siempre de un orden superior a los v_i, w_j ... que figuran en la misma expresión.

"$\vdash$ " para "es teorema"
"$=$df" para "es definido" (véase 2*e*.).

Además usaremos los símbolos del sistema primario como nombres de sí mismo en los siguientes casos:
1. Si aparecen en una fórmula junto con cualquier metasímbolo.
2. Si está entre comillas o en una línea aparte.

B. Algunas veces usaremos como meta-lenguaje un lenguaje simbólico. En este caso también usaremos parte del simbolismo del sistema primario (por ej., en 2*c*. la formulación del Crit. 2-3, y en 4*b*. el MT.4-3.).

Observamos que hay algunos meta-símbolos que son iguales al simbolismo del sistema primario, sin embargo no se producirá confusión en la práctica.

1*d*. Axiomas y reglas axiomáticas.

Mencionamos aquí solamente aquellos axiomas (no son independientes), que aparecerán en las demostraciones posteriores. En paréntesis cuadrados las abreviaciones que usaremos.

a.	axiomas del cálculo proposicional (usuales)	[Prop.]
b.	$(x)\, Fx \supset Fy$	[AxF1.]
c.	$Fy \supset (Ex)\, Fx$	[AxF2.]
c'.	regla de existencialización	[exist.]
d.	$x = x$	[AxIg.1]
e.	$y = x \,.\, z = x \,.\supset.\, y = z$	[AxIg.2.]
f.	axioma de extensionalidad.	[extens.]
g.	Regla de inferencia (o Modus Ponens)	[M. P.]
h.	Regla de sustitución	[Sust. F]

a.	para variables proposicionales	[— — *a*.]
b.	cambio alfabético de variables ligadas	[— — *b*.]
c.	sustitución para variables individuales	[— — *c*.]
d.	sustitución para variables de funciones proposicionales	[— — *d*.]

i. de "$p \supset Fx$" se obtiene "$p \supset (x)Fx$" [Regla F 1.]
con las restricciones usuales.

i'. Regla de la universilación [Univ.]
j. de "$Fx \supset p$" se obtiene "$(Ex)Fx \supset p$" [Regla F 2.]
con las restricciones usuales.

Observamos que hemos formulado los axiomas y las reglas únicamente para un sistema funcional de primer orden. Convenimos en agregar la letra "^{s}", para indicar que se trata del axioma o regla, correspondiente al sistema superior.

Así tenemos por ejemplo:

b^s. $(v)\ Vv \supset Vw$ [AxF 1^s]

con las restricciones usuales.

En todos los casos dejamos libre la posibilidad de usar tanto axiomas formulados en el simbolismo primario (p. ej. *AxF1)*, como también *esquemas axiomáticos* (p. ej. *AxF1*^{s}). Veremos más adelante que trataremos principalmente las definiciones como axiomas, por lo tanto pueden haber definiciones formuladas en lenguaje primario o también formuladas en meta-lenguaje, llamadas *esquemas de definición*. (Nótese la diferencia entre esquemas sentenciales, que se refieren a expresiones del sistema primario, y esquemas axiomáticos (y esquemas de definición), que se refieren a expresiones del meta-lenguaje).

2. TEORIA DE LA DEFINICION

2*a*. Definición formal y semiformal

a) Definición por medio de axiomas en lenguaje primario. En ciertas etapas del desarrollo de un sistema formal se incorporan al conjunto primitivo de símbolos, nuevos símbolos. O también se introducen ordenaciones nuevas de símbolos ya existentes. Para efectuar esto se agregan axiomas nuevos que contienen estos símbolos o las notaciones que se desean introducir. Además se agregan nuevas reglas de formación de expresiones, o se modifican las ya existentes, para incorporar estas nuevas notaciones en el sistema así ampliado.

Si al agregar los axiomas y las reglas, el sistema cumple con las siguientes condiciones hablaremos de *definiciones formales* (respecto a un sistema).

1) *Eliminabilidad.* Debe haber un procedimiento efectivo para eliminar las notaciones nuevas y reemplazarlas por notaciones equivalentes del sistema primitivo.

2) *Conservatividad.* El número de teoremas del sistema antiguo debe permanecer igual, o sea, se mantiene la misma estructura para éste.

Una formulación más precisa se dará en 2*c*.

Ejemplo: si se agrega a un sistema en que se dispone de la constante "*R*" el siguiente axioma,

D2-1 $Px \equiv (E\,y) \quad Rxy$

esto sería una definición formal siempre que se modifiquen convenientemente las reglas de formación.

b) Definición por medio de una expresión en meta-lenguaje. En lugar de introducir las nuevas notaciones por axiomas formulados en lenguaje primario, podemos efectuarlo por medio de una frase en metalenguaje, que sería en este caso una regla axiomática. Se usa generalmente una formulación de un tenor aproximadamente a: " "......." es una notación alternativa de "........" ". Si además se agregan nuevas reglas de formación como en el caso anterior, y si el sistema cumple con las condiciones de conservatividad y eliminabilidad, también hablaremos de una *definición formal* (respecto a un sistema). Por ejemplo:

D2-2. a. " $p \equiv q$ " es una notación alternativa para
" $p \supset q \cdot q \supset p$ "

o la formulación más común,

b. $p \equiv q =_{df} p \supset q \cdot q \supset p$

El meta-símbolo "$=_{df}$" se lee generalmente "es definido por" o "es una abreviación de". (Nosotros adoptaremos más adelante principalmente " son mutuamente derivables " o " se pueden sustituir entre sí ").

c) Definición sin nuevas reglas de formación. Hablamos de una *definición semi-formal* (respecto a un sistema) en el caso que se usa una formulación como en *b*), pero sin que se agreguen reglas de formación, que permitirían incorporar las nuevas notaciones en el sistema formal. En este caso la frase en meta-lenguaje no constituye una regla axiomática, sino permanece fuera del sistema. De tal modo, que cualquiera afirmación sobre el sistema, en que se menciona una fórmula en que aparece la notación nueva, se considerará por convención como si se refiere a la fórmula que resulta al reemplazarla por la notación antigua.

Se usa a veces este método, cuando se quiere evitar complicaciones, al agregar a un sistema formal todo un sistema de definiones. Por otra parte, si en el sistema no se disponen de expresiones del mismo tipo de las notaciones, que se desean introducir, entonces es ésta la forma más cómoda para agregar definiciones. Así, por ejemplo, en un sistema básico de funciones no es posible introducir la siguiente definición formal:

D2-3. $E(x,F) =_{df} Fx$ (x es elemento de F)

ya que no disponemos de predicados cuyos argumentos pueden ser a su vez predicados.

A pesar de estas diferencias, en la práctica las formulaciones de las definiciones de los 3 tipos serán generalmente análogas, siempre que se tome la precaución de reformular adecuadamente las referencias a las reglas de formación en el caso *c)*. Por lo tanto no los tratamos separadamente, excepto algunos casos especiales, sino estudiaremos principalmente la definición formal, o sea, los casos *a)* y *b)*.

En vez de "definición formal" hablaremos en general solamente de "definición", a menos que sea necesario para recalcar ciertos conceptos.

2*b*. Símbolos y notaciones

Frecuentemente en la teoría de la definición se estudia únicamente la definición de símbolos. Aquí trataremos el concepto más amplio de definición de notaciones, de la cual la definición de símbolos será un caso particular.

En una *definición de notación* se introducen en un determinado orden m símbolos nuevos ($m = 0$ o 1), junto con n símbolos antiguos ($n \geqslant 0$). O sea, tenemos las siguientes posibilidades:
a) se puede introducir un símbolo nuevo y aislado sin símbolos antiguos, (en este caso $n = 0$ y $m = 1$).

D2-4. $P =_{df} (\lambda x)\, Qx \vee Rx$
D2-5. $2 =_{df} sig(1)$

b) se puede introducir un símbolo nuevo, junto con un determinado número de símbolos primitivos en un cierto orden (en este caso $m = 1$ y $n > O$).

D2-6. $Pxy =_{df} Qyx$
D2-7. $(E!x)\, Fx =_{df} (Ex)(y) \,.\, Fy \equiv y = x$
D2-8. $G(\iota x)\, Fx =_{df} (E!x) \,.\, Fx \qquad Gx$

c) no se introduce ningún símbolo nuevo sino solamente una ordenación nueva de símbolos primitivos (en este caso $m = O$ y $n > 2$).

D2-9. $xF =_{df} \sim Fx$
D2-10. $pq =_{df} p \,.\, q$
D2-11. $(x)\, Fx =_{df} \sim (Ex) \sim Fx$
D2-12. $x = y = z =_{df} x = y \,.\, y = 2$

Aunque generalmente la intención es incorporar sólo una notación nueva a la vez, sin embargo, por la forma permitida a las definiciones, se introducen a veces simultáneamente diferentes notaciones. Así en D2-8 se incorpora "$G(\iota x)Fx$" a los esquemas sentenciales, "$(\iota x)Fx$" a los esquemas nombrativos (bajo ciertas condiciones), y "ι" a los operadores.

No hemos incluido la definición de 2 o más símbolos nuevos a la vez, ya que ello ocurre raras veces. Daremos aquí únicamente un ejemplo. Supongamos que nuestro lenguaje sea uno que tengan la simbolización de Lukasievicz. (En éste no se necesitan paréntesis para indicar el alcance de las conectivas bi-posicionales, ya que ellos se anteponen a las expresiones). Podemos formular ahora la siguiente definición:

D2-13. $(p \vee q) =_{df} Apq$

por medio de la cual introducimos 3 símbolos, la conectiva "v", y los paréntesis izquierdo y derecho.

2c. Criterio para las definiciones

Daremos ahora una formulación más precisa del criterio que usaremos para considerar una ampliación de un sistema como una definición. Primero la formularemos para una definición de símbolos nuevos, la cual, además, se refiere a un lenguaje determinado. Después la daremos para la definición de notación en general y, en seguida, compararemos ambas formulaciones.

Crit. 2-1. *Criterio de eliminabilidad.*

Una equivalencia C que define un símbolo s satisface el criterio de eliminabilidad si y sólo si, para toda expresión B, en la que figura s hay una expresión A tal, que s no figura en A y $C \cdot \supset \cdot B \equiv A$ es demostrable a partir de los axiomas y definiciones precedentes.

Crit. 2-2. *Criterio de conservatividad.*

Una equivalencia C que define un símbolo s satisface el criterio de conservatividad si y sólo si, no hay una expresión D tal, que s no figura en D y $C \supset D$ es demostrable a partir de los axiomas y definiciones precedentes, pero D no lo es.

Al Crit. 2-1 se le llama también de *retraducibilidad,* ya que según él, exigimos que podamos traducir cada expresión en que aparece el símbolo nuevo, a una en la cual este símbolo no figura. Al Crit. 2-2 también se le dice de *no-creatividad,* ya que no se pueden crear teoremas nuevos, formulables en el simbolismo antiguo.

La siguiente meta-definición nos servirá para formular el criterio para una definición de notación, la cual tomaremos como base para nuestro estudio. Definiremos "$Df\ (S_2, S_1)$" —el sistema S_2 es una definición para el sistema S_1. Usaremos "$Ess\ (A, S_n)$" —A es un esquema sentencial en S_n— y "$Der\ (A, C_1, \ldots, C_n, S_m)$"— A es derivable de $C_1, \ldots, C_n$ en S_m.

Crit. 2-3. $Df\ (S_2, S_1) =_{\text{mdf}} (B) : \cdot\ Ess\ (B, S_1 \vee S_2) \cdot \supset :$
$(EA) . Ess\ (A, S_1) . Der\ (A, B, S_1 \vee S_2) \cdot Der\ (B, A, S_1 \vee S_2) :$ (1)
$(A, C_1, \ldots, C_n) : Ess\ (A \cdot C_1 \ldots C_n, S_1) . \supset . Der\ (A, C_1, \ldots, C_n, S_1 \vee S_2) \equiv Der\ (A, C_1, \ldots, C_n, S_1) : .$ (2)
$(EB) \cdot Ess\ (B, S_1 \vee S_2) \cdot \sim Ess\ (B, S_1).$ (3)

o sea, según (1) exigimos que para cualquier esquema sentencial B del sistema ampliado S_1 v S_2, hay un esquema sentencial A del sistema primitivo S_1 tal, que A y B son mutuamente derivables uno del otro en el sistema ampliado. Según (2), podemos formar una derivación en el sistema ampliado, en la que intervienen sólo expresiones del sistema primitivo, si y sólo si, la podemos formar en el sistema primitivo (o sea, si A se puede derivar de $C_1, \ldots, C_n$ en S_1.), (3) exige que la ampliación introduzca alguna notación nueva, al existir por lo menos un B tal que B sea esquema sentencial en el sistema ampliado y no lo sea en el original. (1) corresponde a eliminabilidad y (2) a conservatividad. (A una ampliación que cumple con (2) se le llama una *extensión conservativa).*

Comparemos ahora las 2 formulaciones del criterio. Llamaremos *a)* a Crit. 2-1 y 2-2, y *b)* a Crit,. 2-3.

a) exige que la definición sea formulada en lenguaje primario y que sea una equivalencia,

b) permite formulaciones en meta-lenguaje, y en el caso de una definición en lenguaje primario puede ser una o varias expresiones de cualquier tipo. (D2-14 a 17).

a) exige que se incorpore algún símbolo nuevo;

b) exige sólo que se incorporen esquemas sentenciales nuevos. *(D2-9* a *12).*

a) deben existir en el sistema los símbolos "$\supset$" y "$\equiv$";

b) lo reemplaza por la relación de derivabilidad. (Sin embargo, esta diferencia es sólo aparente, ya que en la mayoría de los sistemas la derivabilidad corresponde a "$\supset$" y la derivabilidad mutua al "$\equiv$").

a) exige que haya teorema en el sistema;

b) puede haber solamente derivación de premisas. (D2-15). (Sin embargo, vale lo mismo que en el anterior).

Como vemos, lo único que *b)* exige del sistema, es que exista la clase de los esquemas sentenciales y que haya entre algunos de ellos la relación de derivabilidad. Sin embargo, esto no significa que *a)* sea una formulación defectuosa, sino solamente es más restrictiva que *b)*. En la práctica basta generalmente con *a)* para los sistemas usuales.

Ejemplos de definiciones que cumplen con Crit. 2-3 y no con Crit-2-1 y 2-2:

D2-14. $\vdash p \equiv q . \supset . p \supset q \quad q \supset p : p \supset q \cdot q \supset p . \supset . p \equiv q$

En esta definición introducimos "$\equiv$" por medio de un axioma en lenguaje primario. Como vemos, la fórmula no es una equivalencia, pero, sin embargo, cumple con eliminabilidad y conservatividad. (Corresponde a D2-2).

Para el 2º ejemplo, supongamos que tenemos un sistema en que disponemos del símbolo proposicional constante "p_a", y que rige además la siguiente regla axiomática:

$A \vdash A \vee B$ (de A obtenemos $A \vee B$, en que A y B son esquemas sentenciales cualesquiera) [R. Ax. 1].

Si agregamos las siguientes dos reglas, ellas definirían el símbolo proposicional "q_a", de acuerdo a Crit. 2-3,

D2-15.

$p_a \vdash q_a$ [R. Ax. 2]

$q_a \vdash p_a$ [R. Ax. 3]

En este sistema, que corresponde a una parte restringida del cálculo proposicional, no tenemos teoremas, ni los podemos obtener si no agregamos algún símbolo que corresponde al "$\supset$".

Ejemplo Nº 3. Kleene [*Metamathematics*], pág. 408, Teor. 42. Supongamos que de algún predicado "P" podemos demostrar:

$\vdash (E!w)\, P(w, v_1, \ldots, v_n)$ (w, y $v_1 \ldots v_n$ son variables diferentes).

Entonces el siguiente axioma introduce el functor "f_a" en el sistema, cumpliendo con eliminabilidad y conservatividad,

D2-16. $\vdash P(f_a(v_1, \ldots, v_n), v_1, \ldots, v_n)$.

Ejemplo Nº 4. Podemos introducir el predicado "P" por medio del siguiente axioma:

D2-17. $\vdash Px$.

La demostración se dará en 4*d*.

En estos ejemplos se considera que los símbolos introducidos son del tipo correspondiente, y no son símbolos del sistema antiguo, de acuerdo a lo que se tratará en 3*b*.

Hacemos notar que si hubiésemos reemplazado en Crit. 2-3 la eliminabilidad de las notaciones nuevas en los *esquemas sentenciales,* por la eliminabilidad de éstos en las *sentencias,* se podrían haber incluido en este tipo a las definiciones por recurrencia en su formulación usual.

2*d*. Fines de la definición

Debido a las restricciones que hemos impuesto a las definiciones, ellas no constituyen una ampliación *efectiva* del sistema. Las ventajas que se obtienen al introducirlas son principalmente las siguientes:

a) Abreviación. Una definición da la posibilidad de usar una fórmula más corta, en lugar de una expresión que en el sistema original tiene una mayor extensión.

b) Revelación de aspectos implícitos. Por medio de una definición podemos hacer resaltar conceptos que estaban implícitos en una expresión, las cuales a veces después se pueden aislar y usar en forma independiente. Estos conceptos formales corresponden frecuentemente en mayor o menor grado a conceptos intuitivos.

Así, en D2-1 y 2-2 se introducen formulaciones más breves. Además, en D2-2 se introduce el símbolo "$\equiv$", que corresponde al concepto intuitivo de "equivalente". Al usar D2-3 al contrario, resulta muchas veces una formulación más larga, pero por otra parte esta definición introduce el símbolo "E", que corresponde a la relación de "ser elemento". Este símbolo ahora ya se puede usar en formulaciones como "$K(E) \supset M\ (E)$". D2-7 introduce una formulación más corta y que corresponde además al concepto de "hay exactamente un . . .".

Supongamos ahora que nuestro propósito sea demostrar una expresión en un sistema determinado. Podemos entonces intentar la demostración en un sistema, ampliado con definiciones, teniendo a nuestra disposición un mayor número de conceptos, además de reducir frecuentemente la longitud de las fórmulas que intervienen en ella. Por otra parte, cualquiera expresión que es demostrable en el sistema ampliado, corresponde a una expresión demostrable en el sistema original (véase MT 4-5).

El agregar definiciones tampoco nos origina problemas nuevos respecto a la estructura del sistema. Así, por ejemplo, si hemos introducido definiciones a un sistema consistente, éste sigue siendo consistente. Por otra parte, la introducción de axiomas o reglas que no cumplen con conservatividad y eliminabilidad, puede modificar radicalmente el sistema respectivo.

Las definiciones que se introducen tienen generalmente en vista obtener los fines *a)* o *b),* aunque por la liberalidad del Crit. 2-3, podemos formular definiciones que ni son más breves, ni revelan aspectos nuevos.

D2-18. $Px =_{df} Qx$.

Por cierto que en la práctica no habrá interés en introducir este tipo de definiciones, pero formalmente son lícitas.

2*e*. DEFINIENS Y DEFINIENDUM; IMPLÍCITA Y EXPLÍCITA.

Debido a que Crit. 2-3 exige la derivabilidad mutua entre la expresión con notación nueva y algunas otras con simbolización antigua, se usa de preferencia para las definiciones en lenguaje primario, fórmulas que tienen entre otras, la propiedad que el signo principal es "$\equiv$" o "$=$", ya que con ello se puede demostrar esta derivabilidad. En el caso de una definición en meta-lenguaje por medio de una regla axiomática, se usará una formulación análoga que tenga las mismas consecuencias; entre otros "son sustituibles entre sí" o "son mutuamente derivables". Generalmente se adopta entonces el meta-símbolo "$=_{df}$". (También se usa este símbolo en el caso de una definición sin nuevas reglas de formación. (2*a*. caso *c*)).

Nosotros usaremos también de preferencia el "$=_{df}$", pero considerándolo un símbolo ambiguo en el sentido que puede representar tanto al mismo "$=_{df}$" del meta-lenguaje (o sea, como abreviación de "son sustituibles entre sí",... etc.), como también al "$\equiv$" o al "$=$" del lenguaje primario. El contexto indicará generalmente cuál de estos símbolos se quiso usar, en el caso que ello tenga alguna importancia especial. Para distinguir las definiciones con "$=_{df}$" de los otros tipos (D2-14 a 17), las llamaremos a veces *equivalencias* (en sentido amplio).

Llamamos *definiendum* a la notación nueva, y la escribimos a la izquierda del "$=_{df}$" y *definiens* a la expresión en simbolismo primitivo que se escribe a la derecha. La notación nueva se llama *notación definida.* Esta terminología rige también con ciertas modificaciones para el caso qe no se trate de una equivalencia.

En caso que en el definiendum parece un solo símbolo, hablamos de *definición explícita,* en otro caso de *definición implícita.* (Algunos autores llaman a los dos tipos "definición explícita en sentido amplio" (al primero "definición explícita en sentido estricto"), para contraponerlas generalmente a las definiciones por recurrencia).

2*f*. Relaciones con la definición tradicional

Aunque el concepto de definición que hemos determinado aquí tiene puntos de contacto, con lo que tradicionalmente se llamaba definición, sin embargo hay aspectos que lo distinguen. Las condiciones clásicas de la definición corresponden generalmente a las siguientes:

1) No debe ser circular.
2) No debe ser expresada en un lenguaje oscuro o figurativo.
3) No puede estar formulada negativamente.
4) Ella debe dar la esencia de lo que se define.

Si lo comparamos con nuestro criterio, vemos que: 1) aparece indirectamente (véase 3*b*. — Cond. 1*e*). 2) está implícita en la idea de un sistema formal. Condición 3) no se exige, ya que se considera una restricción innecesaria, debido a que en la lógica moderna las expresiones negativas están a la par con las otras. Y 4) no se trata, ya que este problema cae fuera del campo de la lógica.

A pesar que con las condiciones de la definición clásica, ni excluimos 4), se podrían formar definiciones que en gran parte cumplen con nuestro criterio, en la práctica las definiciones clásicas han originado frecuentemente múltiples problemas. Ello se debe especialmente a que en vez de usar símbolos o palabras completamente nuevas, generalmente se introducen por convención palabras que ya tienen una determinada connotación, o palabras derivadas gramaticalmente de otras con una cierta connotación. Y así en la discusión posterior los conceptos implícitos en la expresión definida vuelven a oscurecer los problemas.

El uso en la definición formal de símbolos que en el lenguaje común tienen un cierto significado, tiene por finalidad únicamente sugerir una analogía para facilitar una mayor comprensión intuitiva, sin que este significado pueda intervenir en el desarrollo del sistema.

3. DEFINICION DE PREDICADOS

3*a*. Reglas de formación

En los capítulos siguientes trataremos las condiciones formales para diferentes tipos de definiciones que cumplen con nuestro criterio (Crit. 2-3), entre ellos definiciones de predicados, functores, conectivas, etc. Estas condiciones presentan una gran analogía entre los diversos tipos, por lo cual no será necesario entrar en detalles en todos. En el presente capítulo y en el Nº 4, trataremos, además de las definiciones de predicados, aquellos problemas comunes a las otras definiciones, y se estudiará después únicamente lo específico de las demás.

En una definición podemos distinguir tres etapas. Primero se introduce la notación nueva en el sistema por medio de una regla de formación de expresiones. En seguida se formula el axioma o la regla axiomática. Por último, se debe demostrar que la ampliación del sistema cumple con Crit. 2-3.

Generalmente no se formula una regla de formación nueva en cada definición, sino se prevé en las reglas de formación primitivas la posibilidad de agregar constantes de una categoría determinada, por medio de una definición. Como vemos, en este caso será siempre necesario indicar que la notación definida pertenecerá a tal o tal clase de símbolos o de expresiones. Sin embargo, esto no se hace en la práctica, ya que generalmente por la forma que tiene la expresión definitoria se comprende de inmediato a qué categoría pertenece la notación nueva.

En lo que sigue, nosotros tampoco formularemos una nueva regla de formación en cada definición, sino consideraremos que en las reglas de formación primitiva ya se mencionó la introducción de notaciones definidas. Indicaremos unicamente a qué categoría se incorpora ésta.

3*b*. Condiciones para la definición de predicados.

A pesar que la introducción de una notación definida puede hacerse de varias maneras (D2-14 a 17), nosotros trataremos preferentemente las "equivalencias (en sentido amplio)", ya que las demás definiciones se pueden reducir en gran parte a ésta.

Cond. 3-1. *A.* La expresión que define un predicado *"P"*, del tipo (t_i), tiene la forma

a) $P(v_1, \ldots, v_n) =_{df} Esq$

y además se cumple lo siguiente:

b) *"P"* no es un símbolo primitivo ni ha sido definido previamente.

c) Las únicas variables libres que posiblemente figuran en *Esq* son $v_1 \ldots . v_n$.

d) $v_1 \ldots v_n$ son variables nombrativos diferentes $(n \geqslant O)$.

e) En *Esq* figuran sólo símbolos primitivos o símbolos definidos previamente.

f) Si hay ocurrencia de $v_1 \ldots v_n$ en el definiendum, o sea si $n > O$, entonces $v_1 \ldots, v_n$ debe ser del tipo t_i, y *Esq* es un esquema sentencial o un esquema nombrativo, del tipo (t_i).

g) Si no hay ocurrencias de $v_1 \ldots . v_n$ en el definiendum, o sea si $n = O$, entonces *Esq* es un esquema nombrativo del tipo (t_i).

B. En el caso que junto con la definición se agregan otros axiomas en el sistema ampliado (con o sin la nueva notación), las fórmulas que las eliminan (ver 3*c*) deben ser teoremas del sistema primitivo.

Daremos ahora una lista de definición que cumplen con Cond. 3-1, y que nos servirán para estudiar después su relación con Crit. 2-3. En general en todas las definiciones que daremos suponemos que los símbolos que figuran en el definiens son primitivos, o han sido definidos previamente. Además en los ejemplos matemáticos, los functores que deben ir delante de los argumentos se escribirán a veces de acuerdo al uso habitual, entre estos.

D3-1. a. $Hx =_{df} Ax \cdot Rx$ (*hombre* $=_{df}$ *animal y racional*)
b. $H =_{df} (\lambda x) Ax \cdot Rx$
c. $H =_{df} A \cdot R$

D3-2. a. $Hx =_{df} (Ey)Pyx$ (x es hombre $=_{df}$ hay un y tal que y es padre (humano) de x)
c. $H =_{df} (\lambda x) (Ey)Pyx$
b. $H =_{df} dom_2 (P)$ $(.. =_{df}$ contra-dominio de relación P)

D3-3. a. $x > y =_{df} y < x$
b. $x > y =_{df} (Ez)\, z' + y = x$ (z' es el Nº siguiente a z)

D3-4. $Px =_{df} (Ey)\, x = 2y$ (x es par $=_{df}$, ...)

D3-5. a. $Vx =_{df} x = x$ (x es elemento de la clase universal)
b. $Vx =_{df} (F) \cdot Fx \supset Fx$

D3-6. a. $Rxy =_{df} Px$ (definición arbitraria, véase Proc. 3-3 y Ej. 3-7)
b. $Px =_{df} Q$

3c. COND. 3-1 Y LOS PROCEDIMIENTOS DE ELIMINACIÓN.

La tercera etapa de una definición consiste en la demostración que efectivamente se cumplió con Crit. 2-3 (eliminabilidad y conservatividad). En primer lugar mostraremos que existe un procedimiento efectivo para eliminar las notaciones nuevas en todos los casos. Usaremos generalmente B para la expresión en que aparece el símbolo nuevo, A para la fórmula equivalente con el símbobo ya eliminado (la llamamos *fórmula de eliminación para B*), y C para el definiens.

Definición explícita.

Pro. 3-1 a. Si se define el predicado "*P*" por una definición explícita (ver 2*c*), la eliminación de "*P*" en la expresión B se hace en *forma directa* (salvo el caso previsto en *b.*). Quiere decir se transforma B en A reemplazando todas las ocurrencias de "*P*" por el definiens respectivo.

b. Si $P(Esn_1, \ldots, Esn_n)$ ocurre en B y en $Esn_1 \ldots Esn_n$ hay ocurrencias libres de variables $v_i \ldots v_j$ las cuales figuran ligadas en el definiens C, transformamos primero C en C' por medio de un cambio alfabético para eliminar los $v_i \ldots v_j$ (y así evitar la captura de variables libres). Por último construimos A de B en "forma directa", pero usando en vez del definiens original a C'.

Ej. 3-1. Eliminar "*H*" introducido según D3-1c. en

$(x)Hx \supset Hy$ (*B*).

Ya que no hay peligro de capturar variables libres la eliminación se hace en "forma directa", al reemplazar las ocurrencias de "*H*" por el definiens respectivo.

$(x)(A \cdot R)x \supset (A \cdot R)y$ (*A*).

Ej. 3-2. Eliminar "*H*" introducido según D3-2c. en

$(x)Hx \supset Hy$ (*B*).

Ya que "*x*" e "*y*" figuran libres en algunos (todos) de los argumentos de "*P*" y están ligadas en el definiens:

$(\lambda x)(Ey)Pyx$ (*C*).,

transformados primero C en C' para eliminar "*x*" e "*y*" por medio de un cambio alfabético,

$(\lambda u)(Ez)Pzu$ *(C')*.
Por último construimos en "forma directa"
$(x)((\lambda u)\ (Ez)Pzu)x \supset ((\lambda u)\ (Ez)\ Pzu)y$ *(A)*.

Podemos notar que si no hubiéramos hecho el cambio alfabético en el definiens nos resultaría en vez del teorema *(A)* (el cual se obtiene por sustitución en AxFl) la siguiente expresión:
$(x)\ (Ey)Pyx \supset (Ey)Pyy$ (hemos eliminado el λ —operador), la cual no es teorema. (Traduciendo: Si para todos los x hay un y que es padre de x, entonces hay un y que es padre de y).

Ej. 3-3. Eliminar *"H"* introducido según D3-1. en:
Pa *(B)*
Ya que en *B* no hay ocurrencia de *"H"*, obtenemos:
Pa *(A)* *(=B)*.

Definición implícita.

Proc. 3-2 *a.* *Si* se define *"P"* por medio de una definición implícita (ver 2*c*.), la eliminación de *"P"* se hace del siguiente modo:
Si $P(Esn_1, \ldots, Esn_n)$ ocurre en *B*, y no sucede lo previsto en *b*., transformamos primero el definiens *C* en *D* sustituyendo las ocurrencias de $v_1 \ldots v_m$ $(m \leqslant n$, (ver Cond. 3-1c.)) por $Esn_1 \ldots Esn_m$. Después construimos *A* de *B* reemplazando $P(Esn_1, \ldots, Esn_n)$ por *D*.

b. Si hay posibilidad de capturar variables libres, se hace primero un cambio alfabético en *C* (ver Proc. 3-1b). Después se sigue igual como en *a*.

Ej. 3-4. Eliminar ">" introducido según D3-3a. en
$p \supset x > y$ *(B)*.
Transformamos primero el definiens:
$y < x$ *(C)*
en *D* al reemplazar "x" $(=v_1)$ por "x" $(= Esn_1)$ e "y" $(= Esn_2)$ por "y" $(= v_2)$,
$y < x$ *(D)*.
Por último construimos *A* reemplazando en *B*,
$p \supset y < x$ *(A)*.

Ej. 3-5. Eliminar "P" introducido según D3-4 en

$(x)Px \supset Py$ (B).

Ya que "y" figura libre en uno de los argumentos de "P", efectuamos primero un cambio alfabético en el definiens:

$(Ey)\, x = 2y$ (C)

$(Ez)\, x = 2z$ (C').

Después se construyen los D a partir del definiens C' tal como en el ejemplo anterior. Por último obtenemos A reemplazando los D en B:

$(x)\, (Ex)\, x = 2z \supset (Ez)\, y = 2z$ (A).

Si no se efectúa el cambio alfabético obtenemos

$\ldots\ldots\ldots \supset (Ey)\, y = 2y.$ (?) .

Proc. 3-3. Si se define "P" por medio de una definición implícita y "P" figura en B como argumento de algún predicado o functor (no es posible en un sistema básico de funciones), transformamos primero el definiens C en C'' al anteponer a él un λ —operador con las variables del definiendum en el mismo orden. (En caso que haya variables $v_i \ldots v_j$ que figuran en el definiendum y no en el definiens (ver D3-6), agregamos primero al definiens $\cdot\; v_i = v_i \ldots\ldots v_j = v_j$). En seguida podemos formar A, sustituyendo las ocurrencias de "P" en B en "forma directa por C''".

Ej. 3-6. Eliminar "E" introducido según D2-3 en

$k(E) = M$ (B).

Ya que "E" figura como argumento del functor "k", transformamos primero el definiens C en C'' al anteponer un λ —operador

$(\lambda x, F)Fx$ (C'').

Después construimos en "forma directa"

$k(\lambda x, F)Fx = M$ (A).

Ej. 3-7. Eliminar "R" introducido según D3-6 en

$K(R)$ (B).

Ya que "y" figura en el definiendum y no en el definiens C, transformamos primero C en C^* agregando "$y = y$"

$Px \cdot y = y$ (C^*).

Construimos ahora C'' anteponiendo un λ —operador
$(\lambda x,y)Px \cdot y = y$ (C'').
Por último obtenemos en "forma directa":
$K(\lambda x,y)Px \cdot y = y$ (A).

Proc. 3-3'. Si como en el caso anterior "*P*" figura como argumento, y en el sistema no se dispone del λ —operador, entonces el procedimiento es el siguiente. Transformamos primero *B* en *D*, reemplazando "*P*" en todos los lugares donde no es seguido por su argumento, por una variable funcional del mismo tipo, sea "*F*", la cual por otro lado no aparece en *B*. Después construimos *B'* en la forma siguiente anteponiendo a *D*
$(v_1 \ldots v_n) \cdot F(v_1 \ldots v_n) \equiv P(v_1 \ldots v_n) \cdot \supset \cdot D$ (B').
Esta fórmula es equivalente a *B*.
Después *A* se puede formar a partir de *B'* según Proc. 3-2, ya que "*P*" figura en él con su argumento.

Ej. 3-8. Eliminar "*H*" introducido según D3-1a. en
$Especie(H) \cdot Ha$ (B).
Formamos primero "*D*" al reemplazar "*H*" en los lugares donde figura como argumento, por la variable funcional uniposicional "*F*",
$Especie(F) \cdot Ha$ (D).
Construimos ahora *B'* anteponiendo a *D*,
$(x) \cdot Fx \equiv Hx. \supset \cdot Especie(F) \cdot Ha$ (B').
Por último construimos *A* de *B* según Proc. 3-2
$(x):Fx \cdot \equiv \cdot Ax \cdot Rx: \supset :Especie(F):Ax.Rx$ (A).

Veremos ahora cómo las restricciones que hemos impuesto a Condición 3-1 nos aseguran la eliminabilidad.

Cond. 3-1d. Al exigir que las variables del definiendum sean todas diferentes impide definiciones como:

D?3-1. $Rxx =_{df} Qx \vee Px$

No se puede eliminar "*R*" en una expresión como "*Rab*".

Cond. 3-1e. Evita la "definición en círculo", ya que se pide que en el definiens haya sólo símbolos primitivos (D? 3-2), o definidos previamente (D? 3-3).

D3-2?. $Px =_{df} Qx \,.\, Px$
No se puede eliminar " *P* ".

D?3-3 (i) $Px =_{df} Qx$
D?3-3 (ii) $Qx =_{df} Px$

Tampoco se puede eliminar " *P* ", ya que " *Q* " se define posteriormente.

Cond. 3-1f. Al exigir que $v_1, \ldots, v_n$ tengan el tipo t_i (ya que " *P* " tiene tipo (t_i)), no permite formular definiciones como las siguientes. Supongamos que queremos introducir " *R* " que tiene tipo (0,0), por medio de:

D?3-5. $R(F, x) =_{df} \sim F(x)$

Ya que " *R* " tiene el tipo (0,0), entonces " $R(y,z)$ " es un esquema sentencial; no podemos eliminar " *R* ", ya que " $\sim y(z)$ " no es bien formada.

D?3-5. $R(p,q) =_{df} p \vee q$

No podemos eliminar " *R* " en " *Rab* ".

Como podemos observar en este caso (al igual que en el próximo), la modificación de las reglas de formación debe hacerse también con cierta precaución. Sin embargo, tal como ya lo vimos en 3a. esto no ofrece mayor dificultad.

Cond. 3-1g. Al tratarse de una definición explícita, el definiendum debe ser del mismo tipo que el definiens.

D?3-6. $Q =_{df} a$, en que " *Q* " es del tipo (O), y "*a*" del tipo O .
No podemos eliminar " *Q* " en " $Q(x)$ " ya que " $a(x)$ " no es expresión bien formada.

Las otras restricciones impuestas a Cond. 3-1. son necesarias para asegurar la conservatividad (4*c*).

4. ELIMINABILIDAD Y CONSERVATIVIDAD

4*a*. Meta-teorema de eliminabilidad.

A partir de Cond. 3-1. y Proc. 3-1 a 3-3 podemos ahora formular los siguientes meta-teoremas.

MT4-1. Sea S_1 v S_2 una ampliación del sistema S_1 de acuerdo a Cond. 3-1. y sea B una esquema sentencial cualquiera de S_1 v S_2 (no necesariamente con la notación nueva), entonces:

a. Podemos construir efectivamente una fórmula A tal que,
b. A es un esquema sentencial de S_1,
c. $A \equiv B$ en S_1 v S_2, (o sea $Der(A,B,S_1 vS_2).Der(B,A,S_1 vS_2)$)
d. En A se mantiene el alcance relativo de las conectivas y de los operadores.
e. Si $v_1 \ldots v_m$ son variables libres en A también son libres en B.
f. Si B es un esquema sentencial de S_1, entonces A es la misma fórmula B

Demostración:

a. Ya que según Cond. 3-1d. no hay ambigüedad sobre las variables a sustituir (vea D?3-1.).
b. Proc. 3-1 a 3-3, y Cond. 3-1e, f, g. (vea D?3-2 a 6),
c. Cond. 3-1a, según los procedimientos indicados se hacen siempre sustituciones de equivalentes.
d. Las sustituciones se han hecho en expresiones atómicas.
e. Cond. 3-1c. (Lo inverso no es siempre válido (D3-6.)).
f. Según Cond. 3-1b, *"P"* en este caso no puede figurar en B y por lo tanto la eliminación no se efectúa (Ej. 3-3).

MT4-2. La fórmula de eliminación para el axioma de definición es un teorema.

Según Cond. 3-1a se obtiene por sustitución de "$p \equiv p$". (véase los procedimientos).

4*b*. Meta-teorema de conservatividad.

Formularemos primero un meta-teorema auxiliar, que nos facilitará la demostración de conservatividad.

MT4-3. Si S_1 v S_2 es una ampliación de S_1 de acuerdo a Cond. 3-1, y si $B, D_1 \ldots D_n$ son esquemas sentenciales en S_1 v S_2 y $A, C_1 \ldots C_n$ las fórmulas de eliminación correspondientes a éstos, entonces de

$$Der(B, D_1 \ldots D_n, S_1 vS_2) \supset Der\ (A, C_1 \ldots C_n S_1)^* \qquad (1)$$

obtenemos:

$$Ess(A,C_1 \ldots C_n,S_1). \supset . Der(A,C_1 \ldots C_n,S_1 vS_2) \equiv Der(A,C_1 \ldots C_n S_1) \qquad (2)$$

Demostración: Ya que S_1 vS_2 es una ampliación de S_1 tenemos

$$Der(A,C_1 \ldots C_n \ldots C_n,S_1) \supset Der(A,C_1 \ldots C_n,S_1\ vS_2) \qquad (3)$$

Además ya que según MT4-1*c*: $A,C_1 \ldots C_n \equiv B,D_1 \ldots D_n$ en S_1 vS_2 (4)
podemos reemplazar en el implicado de (3)

$$Der(A,C_1 \ldots C_n,S_1) \supset Der(B,D_1 \ldots D_n,S_1 vS_2) \qquad (5)$$

lo cual junto con (1) queda en

$$Der(B,D_1 \ldots D_n,S_1\ vS_2) \equiv Der(A,C_1 \ldots C_n,S_1) \qquad (6)$$

Ahora en caso que B, $D_1 \ldots D_n$ son esquemas sentenciales del sistema original S_1, entonces las fórmulas de eliminación son de acuerdo a MT4-1f estos mismos, por lo tanto tenemos

$$Ess(B,D_1 \ldots D_n,S_1). \supset . Der(B,D_1 \ldots D_n,S_1\ vS_2) \equiv Der(B,D_1 \ldots D_n S_1)\ (7)\ (= 2)$$

lo cual corresponde al criterio de conservatividad. (Ver 2*c*. Crit. 2-3, fórmula (2) con cambio alfabético).

El siguiente meta-teorema, que es el de conservatividad lo formularemos para un sistema funcional básico. La demostración para un sistema superior de funciones seguiría estas mismas líneas, pero habría que considerar un mayor número de axiomas.

MT4-4. Bajo las mismas condiciones de MT4-3, y si S_1 es un sistema básico de funciones tenemos:

$$Der(B,D_1 \ldots D_n,S_1\ vS_2) \supset Der(A,C_1 \ldots C_n S_1)^*$$

* Traduciendo: Si podemos derivar B de $D_1, \ldots D_n$ en el sistema ampliado S_1 v S_2, entonces podemos derivar A de $C_1 \ldots C_n$ en el sistema original S_1.

Demostración:

I*a*. B se obtiene en una demostración aplicando *SustF*. a algún axioma del cálculo proposicional. (En este caso puede tratarse solamente de SustF*a*. (sustitución de variables proposicionales), ya que SustF *b*., *c* y *d*. (cambio alfabético de variables ligadas, sustitución para variables individuales, sustitución para variables funcionales) no puede presentarse aquí). Entonces también *A* se puede demostrar a partir del mismo axioma, ya que según MT4-1d. en *A* se mantiene el mismo alcance relativo de las conectivas y operadores.

b. *B* se demuestra aplicando SustF *d*. al AxFl. ("$(x)Fx \supset Fy$"). Entonces *A* se puede obtener en la misma forma, ya que según Proc. 3-1 y 3-2 no hay peligro de ligar variables libres (ver Ej. 3-1, 3-2, 3-5). (Aplicación de SustF *b*. o *c*. es trivial, ya que éstas no introducen la notación nueva (MT4-1f)).

c. *B* se demuestra aplicando SustF. a algún axioma adicional del sistema primitivo; *A* se obtiene de la misma manera como en los casos anteriores. (De acuerdo a Cond. 3-1*b*. la notación nueva no puede figurar en este axioma).

d. *B* se demuestra aplicando SustF, a algún axioma que se haya introducido en el sistema ampliado junto con la definición; *A* se demuestra entonces del mismo modo a partir de la fórmula de eliminación que corresponde a este axioma, el cual según Cond. 3-1B es teorema en S_1.

e. *B* se demuestra aplicando SustF. *c*., al axioma de definición. En este caso tenemos dos posibilidades.

1) Si se sustituye alguna variable que aparece tanto en el definiendum como en el definiens, entonces el *A* correspondiente es un teorema demostrable de la misma manera como se trató en MT4-2. (Ver también Ej. 4-1 y 4-5).

2) Si se sustituye algún esquema nombrativo para una variable que figura en el definiendum pero no en el definiens, entonces en la fórmula de eliminación este esquema nombrativo no figura (ver Proc. 3-1 y 3-2), y *A* es el teorema mencionado en MT4-2. (Ej. 4-2).

La SustF *a*. no encuentra aplicación en este caso, ya que según Cond. 3-1c. no pueden figurar variables proposicionales en la definición. (Ver D?3-5).

SustF *b*. es trivial.

SustF *d.* tampoco es posible en un sistema funcional de primer orden; sin embargo podemos notar que la demostración para un sistema superior de funciones seguiría las mismas líneas que para SustF *c.*

II*a.* B se deriva de D (axioma o premisa) por SustF. (usual o restringida). Entonces A se obtiene de C de la misma forma, tal como en casos I*c* y *d.* (Si D es una premisa en S_1 vS_2 entonces C es premisa correspondiente en S_1).

b. B se deriva de D por universilación. Entonces A se obtiene del mismo modo del C respectivo, ya que según MT4-1c. en C no figuran variables que pueden restringir la aplicación de la regla, mientras que ésta fuera permitida en D. (Ver Ej. 4-3 y 4-6). (Lo inverso no es siempre válido. (Ej. 4-4)).

III. B se deriva por regla de inferencia de $D \supset B$ y D.
Podemos distinguir 2 posibilidades:
1) D no contiene la notación definida, en este caso A se puede obtener de $C \supset A$ y C, ya que según MT4d. se mantiene el alcance de las conectivas y operadores en las fórmulas de eliminación.
2) En D figura la notación nueva; en tal caso también podemos obtener A a partir de $C \supset A$ y C, ya que si D es premisa o derivada de premisa, entonces C también lo es en S_1 (II*a.* y *b.*) y si D es teorema entonces según Cond. 3-1B la C correspondiente es teorema en el sistema primitivo. (I y II*b.*). No es posible que se haya derivado D a partir de algún axioma del sistema original, ya que según Cond. 3-1b., la notación definida no aparece en ninguna fórmula de S_1. (Vea Ej. 4-7 y 4-8).

Por lo tanto siempre si podemos formar una derivación en el sistema nuevo, podemos formar una equivalente en el sistema original, con lo cual demostramos MT4-4.

MT4-5. Una ampliación de un sistema funcional de primer orden que se efectúa de acuerdo a Cond. 3-1 cumple con Crit. 2-3. (eliminabilidad, conservatividad, y por lo menos un esquema sentencial nuevo).

Demostración: (1) MT4-1, (2) MT4-4 con 4-3, (3) $P(v_1, . . v_n)$ es un esquema sentencial en S_1 vS_2 y no lo es en S_1 . (Cond. 3-1b).

Los siguientes ejemplos ilustran la demostración de los meta-teoremas anteriores.

Ej. 4-1. Supongamos que ampliamos el sistema S_1 para formar S_1 v S_2 por medio de la siguiente definición,

[Def.] $Rxy =_{df} Qx$ (1)

Podemos formar ahora la siguiente demostración en el sistema ampliado,

[de 1] $\vdash Rxy \equiv Qx$ (2)
[Sust F_a^x, 2] $\vdash Ray \equiv Qa$ (3) $(= B)$

La fórmula de eliminación para (3) es la siguiente:

$Qa \equiv Qa$ (3') $(= A)$

Observamos que (3') es teorema en S_1.

Ej. 4-2. Tal como en el ejemplo anterior, pero cambiando la variable a sustituir,

[Sust F_a^y, 2] $\vdash Rxa \equiv Qx$ (3) $(= B)$

El A correspondiente es teorema en S_1.

$\vdash Qx \equiv Qx$ (3') $(= A)$

Ej. 4-3. Siempre con la misma definición, pero ahora en vez de sustitución, la regla de universilación,

[Premisa] $Rxy \supset Fz$ (2) $(= D)$
[Univ , 2] $Rxy \supset (z) Fz$ ("z" no figura libre en "Rxy") (3) $(= B)$

Podemos formar ahora la misma derivación en el sistema original.

[Premisa en S_1] $Qx \supset Fz$ (2') $(= C)$
[Univ 2'] $Qx \supset (z) Fz$ (siempre "z" no figura libre) (3') $(= A)$

Ej. 4-4. La siguiente derivación no es lícita en S_1 v S_2.

[Premisa en S_1 v S_2]	$Rxy \supset Py$	(2)
[Univ , 2 ?]	$Rxy \supset (y) Py$	(3) ?

Esta derivación no es lícita, ya que "*y*" aparece en el implicante.

Por otra parte, si formamos ahora la derivación con las fórmulas de eliminación en S_1, ella no va contra la regla.

[Premisa]	$Qx \supset Py$	(2′)
[Univ , 2′]	$Qx \supset (y) Py$	(3′)

Sistema superior de funciones. Como se dijo anteriormente, no trataremos en detalle la demostración para el sistema superior de funciones. Mencionaremos aquí solamente el caso para AxIg.1ª y AxIg.2ª.

Ej. 4-4'. Supongamos que se define "P" por medio de la siguiente definición:

$$P =_{df} (\lambda x)\, Qx \qquad (1)$$

Podemos formar ahora las siguientes demostraciones en el sistema ampliado.

a. [Sust. $\frac{-}{P}-$, AxIg. 1ª] $\vdash P = P$ (2)

[Existen. a 2] $\vdash (EH)\, H = P$ (3) $(= B)$

La fórmula de eliminación para (3) es

$$(EH)\, H = (\lambda x)\, Qx \qquad (4)\ (= A)$$

Esta fórmula es teorema en el sistema original.

b. [Sust. en AxIg. 2ª] $\vdash H = P \cdot G = P \cdot \supset \cdot H = G$ (5) $(= B)$

La fórmula de eliminación para (5) es demostrable en el sistema primitivo:

$$\vdash H = (\lambda x)\, Qx \cdot G = (\lambda x)\, Qx \cdot \supset \cdot H = G \qquad (6)\ (= A).$$

Como podemos observar, Cond. 3-1 cumple con conservatividad respecto a estos axiomas, ya que en el sistema superior vale lo siguiente:

MT 4-5′. $(A)\,(E!V)\,(V = (\lambda v_1, \ldots v_n)\,A)$ $(A$ es un esq. sentencial, y V un predicado (variable) de un tipo correspondiente a " $(v_1, \ldots v_n)$."

O sea, podemos asociar a cada esq. sentencial un predicado y éste es único. (Compárese con 6*f*. definición de functores).