

Un modelo de datos, llamado el modelo entidad – relación.

Este modelo incorpora uno de los más importantes conjuntos de información sobre el mundo real.

La vista lógica de datos ha sido un asunto importante en años recientes. Tres grandes modelos de datos han sido propuestos:

El modelo de red, el modelo relacional, el modelo de conjunto de entidades. Estos modelos tienen sus propias fortalezas y debilidades. El modelo de red proporciona una vista más natural de los datos separando entidades y relaciones.

El modelo relacional se basa en la teoría relacional y puede alcanzar un grado más alto de independencia de datos, pero puede perder un poco de la información semántica sobre el mundo real, el modelo de conjunto de entidades que se basa en la teoría de conjuntos, también alcanza un alto grado de independencia de datos, pero se ven los datos como “3” o “rojo” y no puede ser algo natural para las personas. El modelo entidad relación adopta la vista más natural que el mundo real consiste de entidades y relaciones.

Esto incorpora uno de los conjuntos más importantes de información semántica sobre el mundo real

El modelo puede alcanzar un alto grado de independencia de datos y se basan en la teoría relacional y la teoría de conjuntos.

El modelo entidad relación puede ser usado como una base de una vista unificada de datos.

Gran parte del trabajo del pasado ha enfatizado la diferencia entre el modelo de red y el modelo relacional.

## **El Modelo Entidad-Relación**

Vista de datos multinivel

En el estudio del modelo de datos, podemos identificar los niveles de vistas lógicas de los datos como el modelo interesado. Se pueden identificar cuatro niveles de vistas de datos:

1. Información correspondiente a entidades y relaciones que existen en nuestras mentes.
2. Estructura de información donde las entidades y relaciones son representados por datos.
3. Camino independiente de acceso a una estructura de datos, las estructuras de datos que no se encuentran relacionados con esquemas de búsquedas, esquemas indexados, etc.
4. Camino de acceso dependiente de la estructura de datos.

## **Información correspondiente a entidades y relaciones**

En este nivel consideramos entidades y relaciones. Una entidad es una “cosa” que puede ser distintamente identificada. Una persona específica, una compañía, o un evento es un ejemplo de una entidad. Una relación es una asociación entre entidades. Por instancia “padre-hijo” es una relación entre dos “personas” entidades.

La base de datos de una empresa contiene información relevante correspondiente a entidades y relaciones donde la empresa está interesada. Una completa descripción de una entidad o relación puede no ser almacenada o grabada en la base de datos de una empresa.

Es imposible almacenar cada pieza potencial disponible de información sobre entidades y relaciones.

Relación, Rol, y conjunto de relaciones. Considerando asociaciones entre entidades. Un conjunto de relaciones,  $R$ , es una relación matemática entre  $n$  entidades, cada una tomada de un conjunto de entidades:

$$\{[e_1, e_2, \dots, e_n] \mid e_1 \in E_1, e_2 \in E_2, \dots, e_n \in E_n\}$$

Cada tupla de entidades  $[e_1, e_2, \dots, e_n]$ , es una relación.

El rol de una entidad en una relación es la función que realiza en la relación como “Esposo” y “esposa” son roles.

Atributos, valores, conjunto de valores: La información sobre una entidad o una relación se obtiene por observación o medición, y es expresada por un conjunto de pares de atributos valuados.

Un atributo puede ser formalmente definido como una función que mapea desde un conjunto de entidades o un conjunto de relaciones dentro de un conjunto de valores o un producto cartesiano de un conjunto de valores.

$$f: E_i \text{ o } R_i \rightarrow V_i \text{ o } V_1 \times V_2 \times \dots \times V_n$$

Estructura de la información conceptual: Ahora el enfoque es en cómo organizar la información asociada con entidades y relaciones. El método propuesto es separar la información sobre entidades de la información sobre relaciones.

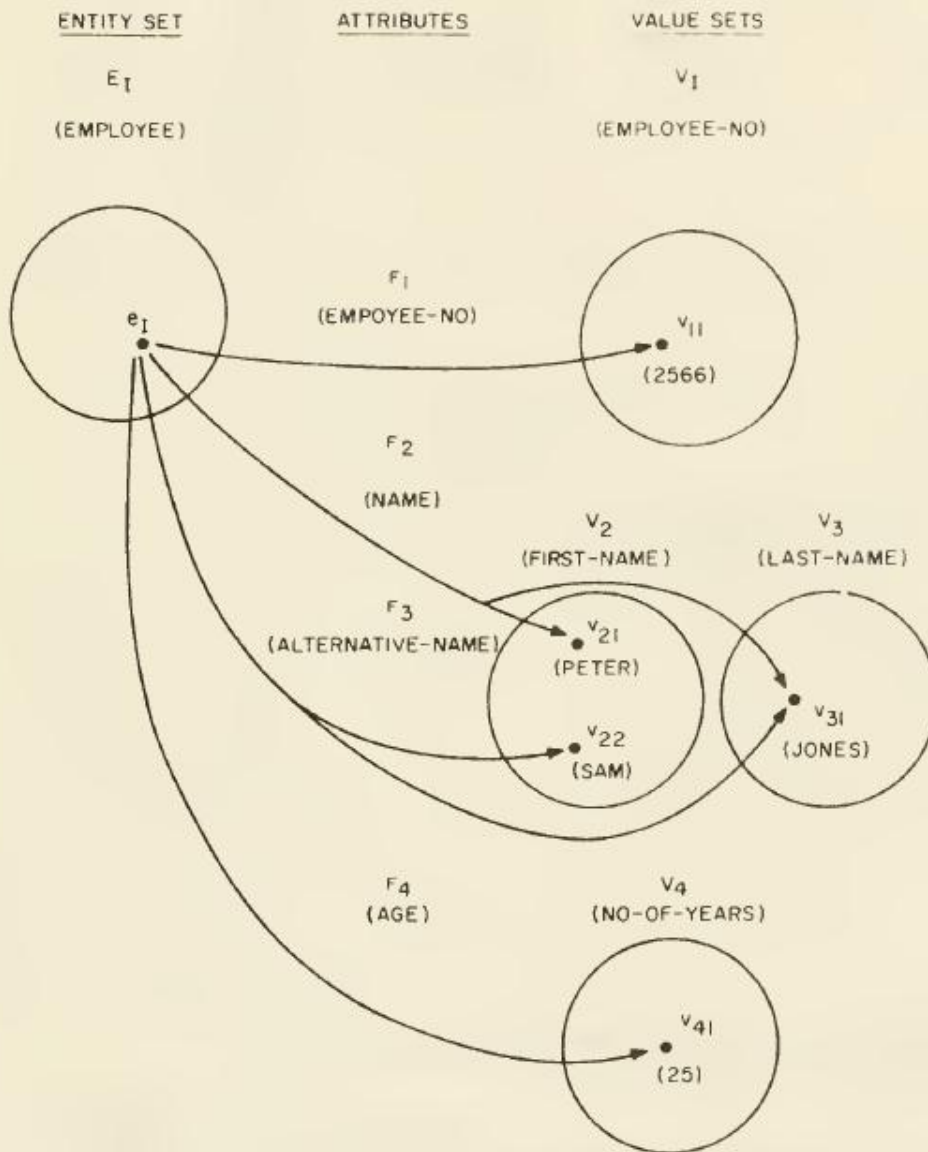


Fig. 2. Attributes defined on the entity set PERSON

Se puede ver que esta separación es útil para identificar dependencias funcionales entre datos.

ATTRIBUTE ENTITY SET AND VALUE SET	F <sub>1</sub> (EMPLOYEE-NO)		F <sub>2</sub> (NAME)		F <sub>3</sub> (ALTERNATIVE-NAME)		F <sub>4</sub> (AGE)
	E <sub>1</sub> (EMPLOYEE)	V <sub>1</sub> (EMPLOYEE-NO)	V <sub>2</sub> (FIRST-NAME)	V <sub>3</sub> (LAST-NAME)	V <sub>2</sub> (FIRST-NAME)	V <sub>3</sub> (LAST-NAME)	V <sub>4</sub> (NO-OF-YEARS)
e <sub>1</sub>		v <sub>11</sub> (2566)	v <sub>21</sub> (PETER)	v <sub>31</sub> (JONES)	v <sub>22</sub> (SAM)	v <sub>31</sub> (JONES)	v <sub>41</sub> (25)
e <sub>2</sub>		v <sub>12</sub> (3378)	v <sub>23</sub> (MARY)	v <sub>32</sub> (CHEN)	v <sub>24</sub> (BARB)	v <sub>33</sub> (CHEN)	v <sub>42</sub> (23)
⋮		⋮	⋮	⋮	⋮	⋮	⋮

Fig. 4. Information about entities in an entity set (table form)

La figura 4 ilustra la tabla de información sobre entidades en un conjunto de entidades. Cada fila de valores está relacionada con la misma entidad, y cada columna está relacionada con el conjunto de valores relacionados con el atributo. El ordenamiento de filas y columnas es insignificante.

ROLE	WORKER		F (PERCENTAGE-OF-TIME)	RELATIONSHIP ATTRIBUTE
	E <sub>1</sub> (EMPLOYEE)	E <sub>J</sub> (PROJECT)		
ENTITY SET	e <sub>11</sub>	e <sub>J1</sub>	v <sub>k</sub> (PERCENTAGE)	VALUE SET
	⋮	⋮	v <sub>k1</sub> (20)	
	⋮	⋮	⋮	

Fig. 5. Information about relationships in a relationship set (table form)

La figura 5 ilustra información sobre relaciones en un conjunto de relaciones. Note que cada fila de valores está afín con una relación que es indicada por un grupo de entidades, cada una tiene un rol específico perteneciendo a un conjunto de entidades específico.

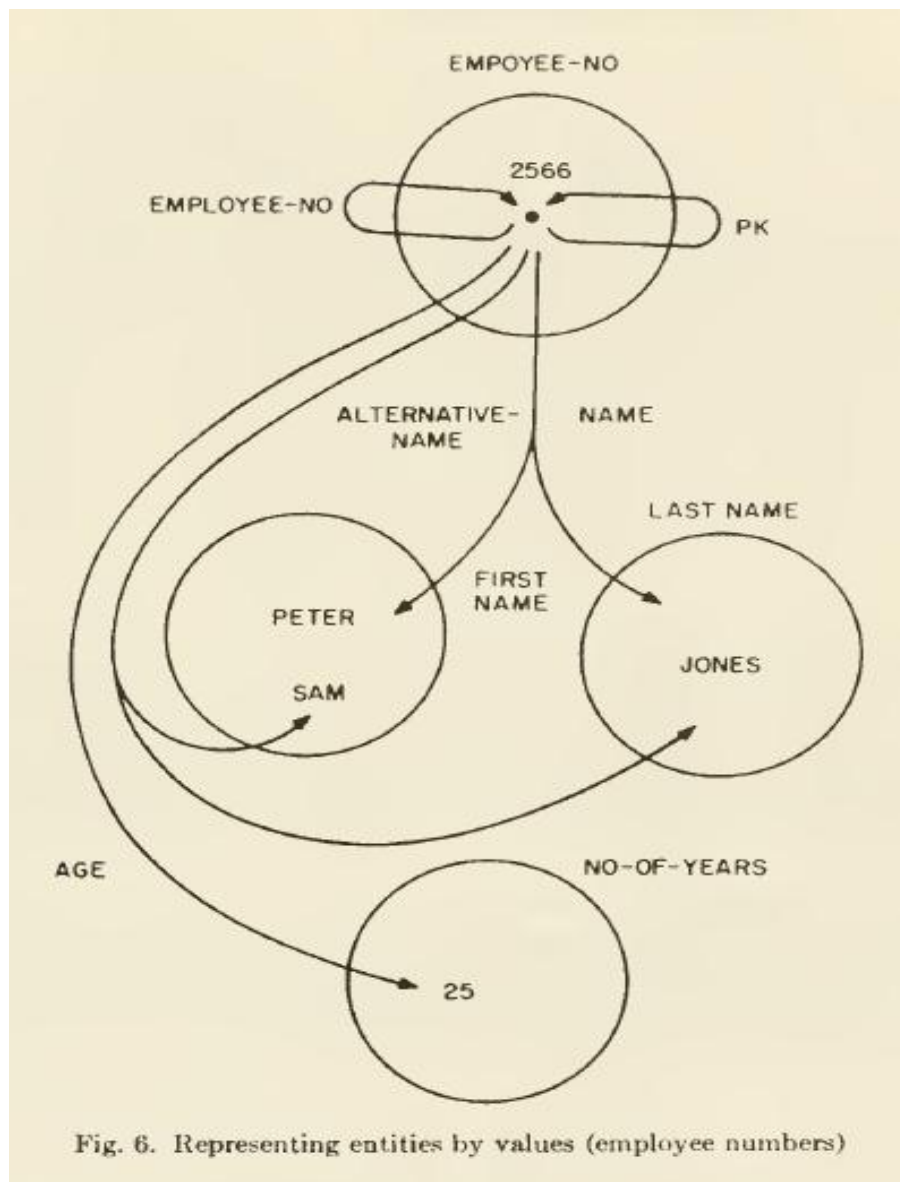
Note que la figura 4 y 2 son formas diferentes de la misma información. La forma de tabla es utilizada por su facilidad de relación para el modelo relacional.

### **Estructura de información (Nivel 2)**

Las entidades, relaciones y los valores en el nivel 1 son objetos conceptuales de nuestra mente, en el nivel 2 se considera representaciones de objetos conceptuales. Asumimos que existen representaciones directas de valores. A continuación se describirá cómo representar entidades y relaciones.

Llave primaria. En la figura 2 los valores del atributo EMPLOYEE-NO pueden ser usados para identificar entidades en el conjunto de entidades de EMPLOYEE si cada empleado tiene un número de empleado diferente. Es posible que más de un atributo sea necesario para identificar las entidades en un conjunto de entidades.

Básicamente una llave de entidad es un grupo de atributos tal que el mapeo es desde el conjunto de entidades hacia el conjunto de valores correspondiente uno a uno. Si no podemos encontrar que sea un mapeo uno a uno en un dato disponible, o su simplicidad en identificar entidades, podemos definir un atributo artificial y un conjunto de valores para que el mapeo sea posible. En el caso donde existan muchas llaves, usualmente elegimos una llave semánticamente significativa como la llave primaria de la entidad.



La Figura 6 se obtiene fusionando el conjunto de entidad EMPLOYEE, con el conjunto de valor EMPLOYEE-NO de la figura 2. Podemos notar algunas implicaciones semánticas de la figura 6. Cada valor en el conjunto de valores EMPLOYEE-NO representa a una entidad (empleado). El mapa de atributos desde el conjunto de valores EMPLOYEE-NO hacia otros conjuntos de valores. También note que el atributo EMPLOYEE-NO mapea desde el conjunto de valor EMPLOYEE-NO hacia si mismo.

		PRIMARY KEY					
ATTRIBUTE	RELATIONAL VIEW	EMPLOYEE-NO	NAME		ALTERNATIVE-NAME	AGE	
VALUE SET (DOMAIN)		EMPLOYEE-NO	FIRST-NAME	LAST-NAME	FIRST-NAME	LAST-NAME	NO-OF-YEARS
ENTITY (TUPLE)		2566	PETER	JONES	SAM	JONES	25
		3378	MARY	CHEN	BARB	CHEN	23
		⋮	⋮	⋮	⋮	⋮	⋮
		⋮	⋮	⋮	⋮	⋮	⋮

Fig. 7. Regular entity relation EMPLOYEE

Relaciones Entidad Relación: información sobre entidades en un conjunto de entidad ahora puede estar organizada de la forma mostrada en la figura 7. Note que la figura 7 es similar a la figura 4 excepto que las entidades son representadas por los valores de sus llaves primarias. Toda la tabla de la figura 7 es una entidad relación y cada fila es una tupla de entidad.

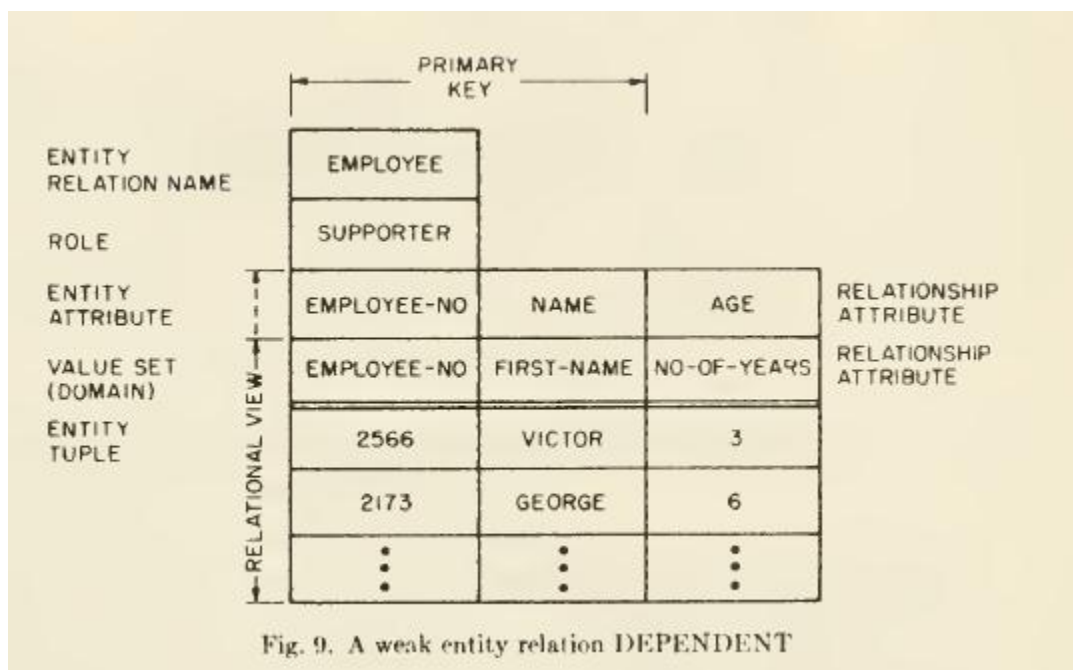
Desde que una relación es identificada por las entidades involucradas, la llave primaria de una relación puede ser representada por las llaves primarias de las entidades involucradas.

	PRIMARY KEY			
ENTITY RELATION NAME	EMPLOYEE	PROJECT		
ROLE	WORKER	PROJECT		
ENTITY ATTRIBUTE	EMPLOYEE-NO	PROJECT-NO	PERCENTAGE-OF-TIME	RELATIONSHIP ATTRIBUTE
VALUE SET (DOMAIN)	EMPLOYEE-NO	PROJECT-NO	PERCENTAGE	
RELATIONSHIP TUPLE	2566	31	20	
	2173	25	100	
	⋮	⋮	⋮	

Fig. 8. Regular relationship relation PROJECT-WORKER

Figura 8. Las entidades involucradas son representadas por sus llaves primarias EMPLOYEE-NO y PROJECT-NO. Los nombres de rol proveen el significado semántico para los valores en las columnas correspondientes. Note que EMPLOYEE-NO es la llave primaria de las entidades involucradas en la relación y no es un atributo de la relación. PERCENTAGE OF TIME es un atributo de la relación. La tabla de la figura 8 es una relación de relación, y cada fila de valores es una tupla de relación.

En ciertos casos, las entidades en un conjunto de entidades no pueden ser únicamente identificadas por los valores de sus propios atributos. Debemos de usar relaciones para identificarlas. Por ejemplo considere dependencias de empleados, las dependencias son identificadas por sus nombres y por los valores de la llave primaria de empleados. Note esto en la figura 9.



EMPLOYEE-NO no es un atributo de una entidad en el conjunto DEPENDENT pero es la llave primaria de employees quien da soporte a los dependientes. Cada fila de valores en la figura 9, es una tupla de una entidad con EMPLOYEE-NO y NAME como su llave primaria, toda la tabla es una entidad relación.

Teóricamente, ningún tipo de relación puede ser usada para identificar entidades. Por simplicidad, podemos restringirnos nosotros mismos al uso de solo un tipo de relación: las relaciones binarias con una cardinalidad 1:N en donde la existencia de n entidades sobre un lado de la relación depende de la existencia de una entidad sobre el otro lado de la relación. Por ejemplo un employee puede tener n ( $=0, 1, 2, \dots$ ) dependientes, y la existencia de dependientes depende de la existencia de un empleado correspondiente.

Este método de identificación de entidades por relaciones con otras entidades pueden ser aplicadas recursivamente hasta que las entidades pueden ser identificadas por sus propios valores de atributos alcanzados. Por ejemplo, la llave primaria de un departamento en una compañía puede consistir en el número de departamento y la llave primaria de la división, en turno consiste del número de división y el nombre de la compañía.



Por lo tanto, tenemos dos formas de relaciones de entidades. Si las relaciones son utilizadas para identificar las entidades entonces podemos llamarlas relación de entidades débiles. Si las relaciones no son utilizadas para identificar las entidades podemos llamarla relación de entidad regular (Figura 7).

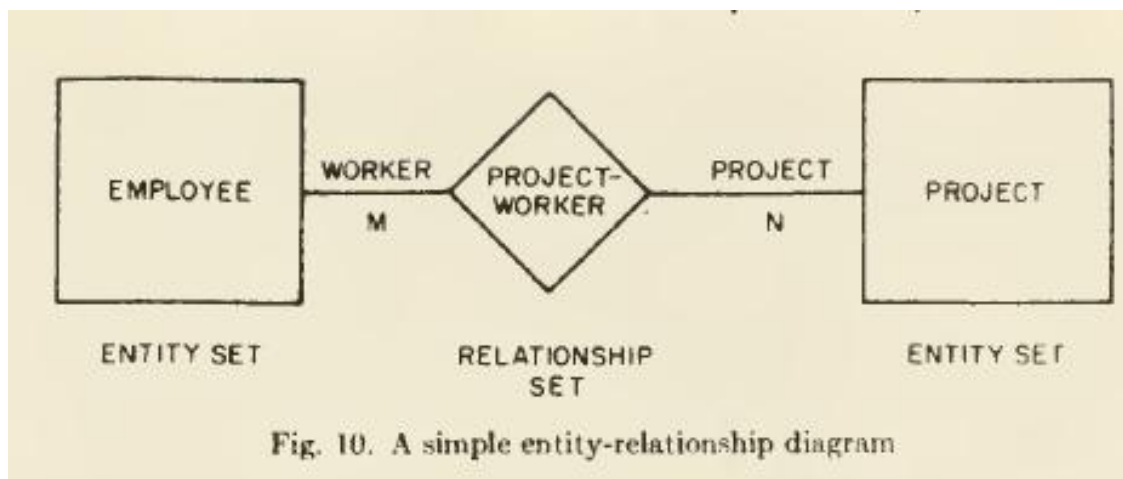
De forma similar tenemos dos formas de relacionar relaciones. Si todas las entidades en la relación son identificadas por sus propios valores de atributos podemos llamarle una relación de relación regular (Figura 8). Si algunas entidades en la relación son identificadas por otras relaciones podemos nombrarla como una relación de relación débil.

Por ejemplo, algunas relaciones entre las entidades de DEPENDENT y otras entidades resultarán en relaciones de relación débiles, desde la entidad DEPENDENT es identificada por su nombre y su relación con la entidad EMPLOYEE. La distinción entre relaciones regulares y relaciones débiles será útil para mantener la integridad de los datos.

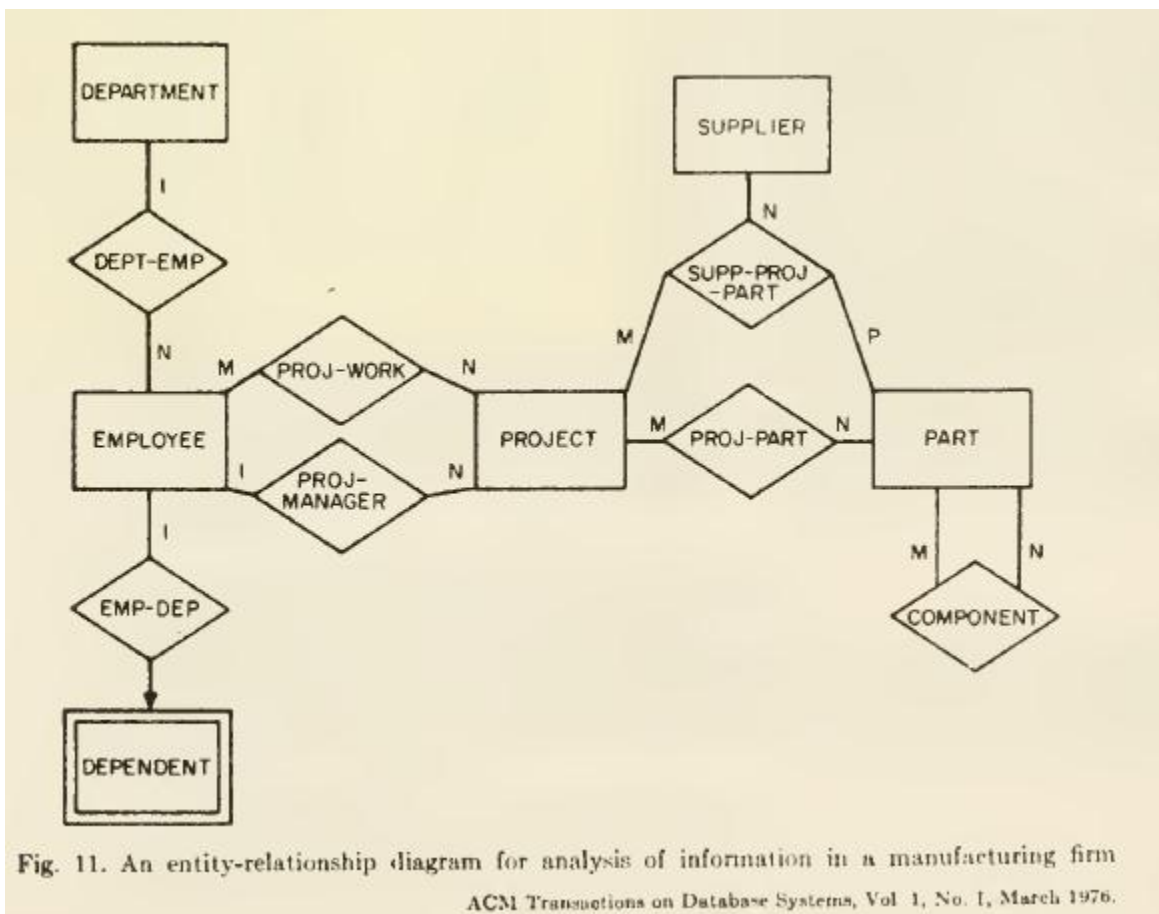
### **Diagrama Entidad – Relación y la inclusión semántica en la descripción y manipulación de los datos.**

#### **Análisis de sistema usando el diagrama Entidad-Relación**

En esta sección introducimos una técnica esquemática para exhibir entidades y relaciones: el diagrama de entidad-relación.



La figura 10 ilustra el conjunto de relaciones PROJECT-WORKER y los conjuntos de entidades EMPLOYEE y PROJECT usando esta técnica esquemática. Cada conjunto de entidades es representado por un rectángulo, y cada conjunto de relación es representado por un figura en forma de diamante. El hecho de que el conjunto de relación PROJECT-WORKER es definido sobre los conjuntos de entidad EMPLOYEE y PROJECT es representado por líneas conectando los rectángulos. Los roles de las entidades en la relación han iniciado.



La figura 11 ilustra un diagrama más completo de algunos conjuntos de entidad y relación que puede ser de interés para una compañía de manufactura.

DEPARTMENT, EMPLOYEE, DEPENDENT, PROJECT, SUPPLIER y PART son conjuntos de entidad.

DEPT-EMP, EMP-DEP, PROJ-WORK, PROJ-MANAGER, SUPP-PROJ-PART, PROJ-PART y COMPONENT son conjuntos de relación.

La relación COMPONENT describe que subpartes (y cantidades) son necesarias para hacer superpartes. El significado de los otros conjuntos de relación necesitan ser explicados.

Muchas características importantes sobre relaciones en general pueden ser encontradas en la figura 11:

- (1) Un conjunto de relación puede ser definido en más de dos conjuntos de entidad. Por ejemplo, el conjunto de relación SUPP-PROJ-PART es definido en tres conjuntos de entidad. SUPPLIER, PROJECT y PART.
- (2) Un conjunto de relación puede ser definido en un único conjunto de entidad. Por ejemplo el conjunto de relación COMPONENT es definido sobre un conjunto de entidad PART.
- (3) Puede que haya más de un conjunto de relación definido sobre conjuntos de entidad dados. Por ejemplo los conjuntos de relación PROJ-WORK y PROJ-MANAGER son definidos sobre los conjuntos de entidad PROJECT y EMPLOYEE.

- (4) El diagrama puede distinguir entre cardinalidad 1:N, M:N y 1:1. El conjunto de relación DEPT-EMP es una cardinalidad 1:N que es que un departamento puede tener  $n$  ( $n = 0, 1, 2, \dots$ ) empleados y que cada empleado trabaja para solo un departamento. El conjunto de relación PROJ-WORK es una cardinalidad M:N donde cada proyecto puede tener cero, uno o más empleados asignados y que cada empleado puede ser asignado a cero, uno o más proyectos.

También es posible expresar la cardinalidad 1:1 del conjunto de relación MARRIAGE.

La información sobre el número de entidades en cada conjunto de entidad donde sea permitido en un conjunto de relación indicado, especificando "1", "m", "n" en el diagrama. El modelo relacional y el modelo de conjunto de entidad no incluyen este tipo de información. El modelo de red no puede expresar una cardinalidad 1:1 fácilmente

- (5) El diagrama puede expresar la dependencia de existencia de un tipo de entidad sobre otra. Por ejemplo, la flecha en el conjunto de relación EMP-DEP indica la existencia de una entidad en el conjunto de entidad DEPENDENT depende de la entidad correspondiente en el conjunto de entidad EMPLOYEE. Si un empleado deja la compañía, sus dependencias ya no serán de interés.

Note que el conjunto de entidad DEPENDENT es mostrado como un rectángulo especial. Esto indica que en el nivel 2 la información sobre entidades en este conjunto está organizada como como una entidad débil (usando la llave primaria de EMPLOYEE como parte de su llave primaria).

### **Un ejemplo de diseño de una base de datos y descripción**

Hay cuatro pasos para diseñar una base de datos usando el modelo entidad-relación:

1. Identifica los conjuntos de entidad y los conjuntos de relación de interés.
2. Identifica la información semántica en los conjuntos de relación como si un cierto conjunto de relación tiene una cardinalidad 1:N.
3. Define los conjuntos de valores y atributos.
4. Organiza los datos en entidades/relaciones y decide llaves primarias.

Los resultados de los primeros dos pasos del diseño de una base de datos están expresados en el modelo entidad-relación como vemos en la figura 11. El tercer paso es definir conjuntos de valores y atributos. El cuarto paso es decidir las llaves primarias de las entidades y las relaciones y organizar los datos como entidad/relaciones de relación. Note que cada conjunto de entidad/relación en la figura 11 tiene una relación correspondiente de entidad-relación. Podemos usar los nombres de los conjuntos de entidad como los nombres correspondientes de las relaciones de entidad/relación tan pronto como no haya confusiones.

Al final de la sección ilustramos un esquema (definición de datos) por una pequeña parte de la base de datos sobre el ejemplo de la compañía manufacturera (la sintaxis de la definición de datos no es importante). Note que el conjunto de valores son definidos con especificaciones de representaciones y valores permitidos. Por ejemplo valores en EMPLOYEE-NO son representados como enteros de 4 dígitos de un rango de 0 a 2000.

Después declaramos tres entidades relaciones: EMPLOYEE, PROJECT y DEPENDENT. Los conjuntos de valores y atributos definidos sobre conjuntos de entidad así como las llaves primarias son seleccionados.

DEPENDENT es una entidad débil desde que utiliza EMPLOYEE.PK como parte de su llave primaria. Podemos declarar dos relaciones: PROJ-WORK y EMP-DEP. Los roles y las entidades involucradas en las relaciones son especificadas. Usamos EMPLOYEE.PK para indicar el nombre de la entidad (EMPLOYEE) y cualquier conjunto de valor-atributo usados en las llaves primarias en esa entidad. El número máximo de entidades de un conjunto de entidad en una relación iniciada. Por ejemplo, PROJ-WORK tiene una cardinalidad M:N. Podemos especificar los valores de M y N. Podemos especificar también el número mínimo de entidades en adición al número máximo. EMP-DEP es una relación débil desde una relación con la entidad, DEPENDENT, es una entidad débil.

<u>DECLARE</u>	<u>VALUE-SETS</u>	<u>REPRESENTATION</u>	<u>ALLOWABLE-VALUES</u>
	EMPLOYEE-NO	INTEGER (4)	(0,2000)
	FIRST-NAME	CHARACTER (8)	ALL
	LAST-NAME	CHARACTER (10)	ALL
	NO-OF-YEARS	INTEGER (3)	(0,100)
	PROJECT-NO	INTEGER (3)	(1,500)
	PERCENTAGE	FIXED (5.2)	(0,100.00)
<u>DECLARE</u>	<u>REGULAR ENTITY RELATION EMPLOYEE</u>		
	<u>ATTRIBUTE/VALUE-SET:</u>		
	EMPLOYEE-NO/EMPLOYEE-NO		
	NAME/(FIRST-NAME, LAST-NAME)		
	ALTERNATIVE-NAME/(FIRST-NAME, LAST-NAME)		
	AGE/NO-OF-YEARS		
	<u>PRIMARY KEY:</u>		
	EMPLOYEE-NO		

<u>DECLARE</u>	<u>REGULAR ENTITY RELATION PROJECT</u> <u>ATTRIBUTE/VALUE-SET:</u> PROJECT-NO/PROJECT-NO <u>PRIMARY KEY:</u> PROJECT-NO
<u>DECLARE</u>	<u>REGULAR RELATIONSHIP RELATION PROJECT-WORKER</u> <u>ROLE/ENTITY-RELATION.PK/MAX-NO-OF-ENTITIES</u> WORKER/EMPLOYEE.PK/m PROJECT/PROJECT.PK/n (m:n mapping) <u>ATTRIBUTE/VALUE-SET:</u> PERCENTAGE-OF-TIME/PERCENTAGE
<u>DECLARE</u>	<u>WEAK RELATIONSHIP RELATION EMPLOYEE-DEPENDENT</u> <u>ROLE/ENTITY-RELATION.PK/MAX-NO-OF-ENTITIES</u> SUPPORTER/EMPLOYEE.PK/1 DEPENDENT/DEPENDENT.PK/n <u>EXISTENCE OF DEPENDENT DEPENDS ON</u> <u>EXISTENCE OF SUPPORTER</u>
<u>DECLARE</u>	<u>WEAK ENTITY RELATION DEPENDENT</u> <u>ATTRIBUTE/VALUE-SET:</u> NAME/FIRST-NAME AGE/NO-OF-YEARS <u>PRIMARY KEY:</u> NAME EMPLOYEE.PK <u>THROUGH</u> EMPLOYEE-DEPENDENT

## Implicaciones sobre la Integridad de datos

Algún trabajo ha sido hecho sobre la integridad de datos para otros modelos. Con conceptos explícitos de entidad y relación, el modelo entidad-relación será útil en entender y especificar restricciones para el mantenimiento de la integridad de datos.

Por ejemplo, hay tres tipos principales de restricciones respecto a los valores:

- (1) Restricciones sobre valores permitidos para un conjunto de valores. Este punto fue discutido en la definición del esquema.
- (2) Restricciones sobre valores permitidos para un cierto atributo. En algunos casos no todos los valores permitidos en un conjunto de valores son permitidos para algunos atributos. Por ejemplo, podemos tener una restricción de edades de empleados entre 20 y 65 años. Esto es:

$$AGE(e) \in (20,65), \text{ where } e \in EMPLOYEE$$

Note que utilizamos notaciones de nivel 1 para clarificar la semántica. Desde que cada conjunto de entidad/relación tiene una relación entidad/relación correspondiente. La expresión puede ser fácilmente traducida en una notación de nivel 2.

(3) Restricciones sobre valores existentes en la base de datos. Hay dos tipos de restricciones:

(i) Restricciones entre conjuntos de valores existentes. Por ejemplo:

$$\{NAME(e) \mid e \in MALE - PERSON\} \subseteq \{NAME(e) \mid e \in PERSON\}$$

(ii) Restricciones entre valores particulares. Por ejemplo,

$$TAX(e) \leq SALARY(e), e \in EMPLOYEE$$

$$BUDGET(e_1) = \sum BUDGET(e_2), \text{ where } e_1 \in COMPANY$$

$$e_2 \in DEPARTMENT$$

$$\text{and } [e_1, e_2] \in COMPANY - DEPARTMENT$$

### Semántica y conjunto de operaciones de consultas de recuperación de información

La semántica y el conjunto de operaciones de consultas de recuperación de información se han vuelto muy claros si las consultas se basan en el modelo entidad-relación de datos. Por claridad, primero discutimos la situación del nivel 1. Conceptualmente los elementos de información están organizados como en la figura 4 o 5. Muchas de las consultas de información pueden ser consideradas como una combinación de los tipos de operación básica:

1. Selección de un subconjunto de valores desde un conjunto de valor.
2. Selección de un subconjunto de entidades desde un conjunto de entidad. Las entidades son seleccionadas por los valores elegidos para ciertos atributos y/o sus relaciones con otras entidades.
3. Selección de un subconjunto de relaciones desde un conjunto de relación. Las relaciones son seleccionadas por los valores elegidos de ciertos atributos y/o por la identificación de ciertas entidades en la relación.
4. Selección de un subconjunto de atributos.

Una solicitud de recuperación de información como “¿Cuáles son las edades de los empleados cuyos pesos son mayores a 170 y quienes fueron asignados al proyecto con PROJECT-NO 254?” puede ser expresada como:

$$\{AGE(e) \mid e \in EMPLOYEE, WEIGHT(e) > 170,$$

$$[e, e_i] \in PROJECT - WORKER, e_i \in PROJECT,$$

$$PROJECT - NO(e_i) = 254\}$$

O

$$\{AGE(EMPLOYEE) \mid WEIGHT(EMPLOYEE) > 170,$$

$$[EMPLOYEE, PROJECT] \in PROJECT - WORKER,$$

$$PROJECT - NO(EMPLOYEE) = 254\}$$

Para recuperar información como la Figura 6 del nivel 2, “entidades” y “relaciones” en (2) y (3) deberían ser reemplazadas por “entidad PK” y “relación PK”. La consulta de recuperación de información puede ser expresada como:

$\{AGE(EMPLOYEE.PK) \mid WEIGHT(EMPLOYEE.PK) > 170$

$(WORKER/EMPLOYEE.PK, PROJECT/PROJECT.PK) \in \{PROJECT - WORKER.PK\},$

$PROJECT - NO (PROJECT.PK) = 254\}$

Para recuperar información como está organizada en las relaciones entidad/relación (Figura 7, 8 y 9), podemos expresarlo como en lenguaje SEQUEL:

SELECT AGE

FROM EMPLOYEE

WHERE WEIGHT > 170

Tabla 1. Inserción

Nivel 1	Nivel 2
Operación: Inserta una entidad en un conjunto de entidades	Operación: Crea una tupla de entidad con cierta llave primaria de entidad Checa: Si la llave primaria (PK) ya existe o si es aceptable.
Operación: Inserta una relación en el conjunto de relaciones Checa: Si las entidades existen	Operación: Crea una tupla de relación con la llave primaria de la entidad dada Checa: Si la llave primaria de la entidad existe
Operación: Inserta propiedades de una entidad o una relación Checa: Si el valor es aceptable	Operación: Inserta valores dentro de la tupla de la entidad o en la tupla de relación Checa: Si los valores son aceptables

AND EMPLOYEE.PK=

SELECT      WORKER/EMPLOYEE.PK

FROM        PROJECT-WORKER

WHERE       PROJECT-NO = 254

Tabla 2. Actualizar

Nivel 1	Nivel 2
Operación: Cambia el valor de un atributo de una entidad	Operación: Actualiza un valor Consecuencia: <ul style="list-style-type: none"> <li>• Si no es parte de una llave primaria de una entidad entonces no hay consecuencias</li> <li>• Si es parte de una llave primaria de una entidad <ul style="list-style-type: none"> <li>** Cambia la llave primaria de la entidad en todas las relaciones relacionadas.</li> <li>** Cambia llaves primarias de otras entidades que usan este valor como parte de sus llaves primarias, por ejemplo , las llaves primarias de DEPENDENTS usa la llave primaria de EMPLOYEES.</li> </ul> </li> </ul>
Operación: Cambia el valor de un atributo de una relación	Operación: Actualiza un valor (Note que un atributo de una relación no será una llave primaria de la relación.

Tabla 3. Eliminación

Nivel 1	Nivel 2
Operación: Elimina una entidad Consecuencias: <ul style="list-style-type: none"> <li>• Elimina alguna entidad cuya existencia dependa de esta entidad</li> </ul>	Operación: Elimina una una tupla de la entidad Consecuencias( aplicadas recursivamente): <ul style="list-style-type: none"> <li>• Elimina alguna tupla de entidad cuya existencia dependa de esta tupla de entidad</li> <li>• Elimina tuplas de relación asociadas con esta entidad</li> </ul>
Operación: Elimina una relación Consecuencias: Elimina todas las propiedades relacionadas	Operación: Elimina una tupla de relación

Es posible recuperar información sobre entidades en dos conjuntos de entidades diferentes sin especificar una relación entre ellos. Por ejemplo, una solicitud de recuperación de información como “Enlista los nombres de empleados y relaciona cual tiene la misma edad” puede expresarse como notación de nivel 1:

$$\{(NAME(e_1), NAME(e_2)) \mid e_1 \in SHIP, AGE(e_1) = AGE(e_2)\}$$

No se discutirá aquí la sintaxis del lenguaje. Lo que queremos mostrar es que las consultas de información pueden ser expresadas usando notaciones de conjuntos y operaciones de conjuntos, y las semánticas de las consultas son muy claras en adoptar un punto de vista.



## Semántica y reglas para inserción, eliminación y actualización

Siempre es un problema difícil mantener la consistencia de datos como inserción, eliminación y actualización de datos en la base de datos. Una de las razones principales es que la semántica y consecuencias de las operaciones de inserción, eliminación y actualización usualmente no son claramente definidas; Esto es difícil de encontrar en un conjunto de reglas que pueden fortalecer la consistencia de datos. Podemos ver esta consistencia de datos se simplifica usando el modelo entidad-relación.

En las tablas 1 a 3, discutimos las semánticas y reglas de inserción, eliminación, y actualización en ambos niveles, nivel 1 y nivel 2, El nivel 1 es utilizado para clarificar la semántica.

## Análisis de otros modelos de datos y su derivación del modelo entidad-relación

### El modelo relacional

La vista relacional de datos y la ambigüedad de la semántica, en el modelo relacional, relación, R, es una relación matemática definida en los conjuntos,  $X_1, X_2, \dots, X_n$ :

$$R = \{(x_1, x_2, \dots, x_n) \mid x_1 \in X_1, x_2 \in X_2, \dots, x_n \in X_n\}$$

El conjunto  $X_1, X_2, \dots, X_n$  son llamados dominios y  $(x_1, x_2, \dots, x_n)$  es llamado tupla. La figura 12 ilustra una relación llamada EMPLOYEE.

ROLE		LEGAL	LEGAL	ALTERNATIVE	ALTERNATIVE	
DOMAIN	EMPLOYEE- NO	FIRST- NAME	LAST- NAME	FIRST- NAME	LAST- NAME	NO-OF- YEARS
TUPLE	2566	PETER	JONES	SAM	JONES	25
	3378	MARY	CHEN	BARB	CHEN	23

Fig. 12. Relation EMPLOYEE

Los dominios en la relación son EMPLOYEE NO, FIRST-NAME, LAST NAME, FIRST NAME, LAST NAME, NO OF YEAR. El orden de filas y columnas en la relación no tienen relevancia. Para evadir la ambigüedad de columnas con el mismo dominio en la relación, los dominios se califican por roles (para distinguir el rol del dominio en la relación). Por ejemplo, en la relación EMPLOYEE, los dominios FIRST NAME y LAST NAME pueden ser calificados por roles LEGAL o ALTERNATIVE. Un nombre de atributo en el modelo relacional es un nombre de dominio concatenado con un nombre de rol.

Comparando la Figura 12 con la Figura 7, podemos ver que los “dominios” son básicamente equivalente a un conjunto de valores. Aunque “rol” o “atributo” en el modelo relacional puedan verse que sirven al mismo propósito como “atributo” en el modelo entidad-relación, las semánticas de estas condiciones son diferentes. El “rol” o “atributo” en el modelo relacional es principalmente utilizado para distinguir dominios con el mismo nombre en la misma relación, mientras “atributo” en el modelo entidad-relación es una función que cardinaliza desde un conjunto de entidad (o relación) en un conjunto de valores.

PROJECT-NO	EMPLOYEE-NO
7	2566
3	2566
7	3378

Fig. 13. Relation EMPLOYEE-PROJECT

Usando operadores relacionales en el modelo relacional puede causar ambigüedades de semántica. Por ejemplo, la unión de la relación EMPLOYEE con la relación EMPLOYEE-PROJECT (Figura 13) en el dominio EMPLOYEE-NO produce la relación EMPLOYEE-PROJECT' (Figura 14).

		LEGAL	LEGAL	ALTERNATIVE	ALTERNATIVE	NO-OF-YEARS
PROJECT-NO	EMPLOYEE-NO	FIRST-NAME	LAST-NAME	FIRST-NAME	LAST-NAME	
7	2566	PETER	JONES	SAM	JONES	25
3	2566	PETER	JONES	SAM	JONES	25
7	3378	MARY	CHEN	BARB	CHEN	23

Fig. 14. Relation EMPLOYEE-PROJECT' as a "join" of relations EMPLOYEE and EMPLOYEE-PROJECT

Pero, ¿cuál es el significado de una unión entre la relación EMPLOYEE con la relación SHIP en el dominio NO-OF-YEARS (Figura 15)? El problema es que el mismo nombre de dominio puede tener diferentes semánticas en diferentes relaciones (note que un rol está destinado a distinguir dominios en una relación dada, no en todas las relaciones). Si el dominio NO-OF-YEAR de la relación EMPLOYEE no está permitida ser comparada con el dominio NO-OF-YEAR de la relación SHIP, diferentes nombres de dominio tienen que ser declarados.

Pero tal comparación es aceptable, el sistema de la base de datos puede advertir al usuario.

En el modelo entidad-relación, las semánticas del dato son mucho más aparentes.

Por ejemplo, una columna en el ejemplo puesto contiene los valores de AGE de SHIP. Si esta información semántica es expuesta al usuario, el operara cautelosamente (refiriéndose a la muestra de información de la consulta de recuperación). Desde que el sistema de base de datos contiene la información semántica, puede ser posible advertir al usuario de problemas potenciales para una operación “como unión”.

SHIP-NO	NAME	NO-OF-YEARS
037	MISSOURI	25
056	VIRGINIA	10

Fig. 15. Relation SHIP

Semánticas de Dependencias Funcionales Entre Datos. En el modelo relacional, “atributo” B de una relación es funcionalmente dependiente de “atributo” A de la misma relación si cada valor de A tiene más de un valor de B asociado el en la misma relación. La semántica de dependencias funcionales entre datos se hace más clara en el modelo entidad-relación. Básicamente, hay dos tipos principales de dependencias funcionales:

(1) Dependencias funcionales basadas en la descripción de entidades o relaciones.

Desde de que un atributo es definido como una función. Este cardinaliza una entidad en un conjunto de entidad hacia un valor singular en un conjunto de valores (ver Figura 2). En el nivel 2, los valores de la llave primaria son usados para representar entidades. Por lo tanto, conjuntos de mono valores (dominios) son funcionalmente dependientes de la llave primaria de un conjunto de valores (por ejemplo, en la Figura 6 y 7, NO-OF-YEARS es funcionalmente dependiente de EMPLOYEE-NO). Desde que una relación puede tener muchas llaves, los conjuntos de mono valores funcionalmente dependerán de una llave de un conjunto de valor. Los conjuntos de valores llave serán mutuamente funcionalmente dependientes de cada una. Similarmente, los conjuntos de mono valores serán funcionalmente dependientes de conjuntos de valor de llave primaria (por ejemplo, en la Figura 8, PERCENTAGE es funcionalmente dependiente de EMPLOYEE-NO y PROJECT-NO).

- (2) Dependencias funcionales basadas en entidades dentro de una relación. Note que en la Figura 11 identificamos los tipos de cardinalidades (1:N, M:N, etc.) para conjuntos de relación. Por ejemplo PROJECT-MANAGER tiene una cardinalidad 1:N. Asumamos que PROJECT-NO es la llave primaria en la entidad relación PROJECT. En la relación PROJECT-MANAGER, el conjunto de valor EMPLOYEE-NO será funcionalmente dependiente del conjunto de valor PROJECT-NO (cada proyecto tiene solo un manejador).

La distinción entre Nivel 1 (Figura 2) y nivel 2 (Figura 6 y 7) y la separación de la entidad relación (Figura 7) de una relación (Figura 8) clarifica la semántica de las dependencias funcionales entre datos.

Relaciones de 3ª Forma Normal versus Relaciones Entidad-Relación. De la definición de “relación”, cualquier agrupación de dominios puede ser considerada para ser una relación. Para evadir propiedades indeseables en el mantenimiento de relaciones, un proceso de normalización es propuesto para transformar relaciones arbitrarias en la primera forma normal, después en la segunda forma normal y finalmente en la tercera forma normal (3FN). Podemos mostrar que la relación entidad-relación en el modelo entidad-relación son similares a las relaciones de la 3FN pero con semántica con más claridad y sin utilizar las operaciones de transformación.

Usemos una versión simplificada de un ejemplo de normalización.

Las siguientes tres relaciones están en la primera forma normal (no hay dominio cuyos elementos están relacionados consigo mismos):

EMPLOYEE (EMPLOYEE-NO)

PART (PART-NO, PART-DESCRIPTION, QUANTITY-ON-HAND)

PART-PROJECT(PART-NO, PROJECT-NO, PROJECT-DESCRIPTION,  
PROJECT-MANAGER-NO, QUANTITY-COMMITTED).

Note que el dominio de PROJECT-MANAGER-NO actualmente contiene el PROJECT-NO del manejador del proyecto. En la relación anterior las llaves primarias están subrayadas.

Ciertas reglas son aplicadas para transformar las relaciones anteriores en la tercera forma normal:

EMPLOYEE' (EMPLOYEE-NO)

PART' (PART-NO, PART-DESCRIPTION, QUANTITY-ON-HAND)

PROJECT' (PROJECT-NO, PROJECT-DESCRIPTION, PROJECT-MANAGER-NO)

PART-PROJECT' (PART-NO, PROJECT-NO, QUANTITY-COMMITTED)

Usando el diagrama entidad-relación en la Figura 11, la siguiente entidad y las relaciones pueden ser fácilmente derivadas:

Entidad        PART" (PART-NO, PART-DESCRIPTION, QUANTITY-ON-HAND)

Relaciones    PROJECT" (PROJECT-NO, PROJECT-DESCRIPTION)

EMPLOYEE" (EMPLOYEE-NO)

Relación de relaciones PART-PROJECT" (PART/PART-NO, PROJECT/PROJECT-NO,

QUANTITY-COMMITTED)

PROJECT-MANAGER"(PROJECT/PROJECT-NO,

MANAGER/EMPLOYEE-NO)

Los nombres de rol de las entidades en relaciones (como MANAGER) son indicadas.

Los nombres de la entidad relación asociados con las llaves primarias de entidades en relaciones y los nombres del conjunto de valor han sido omitidos.

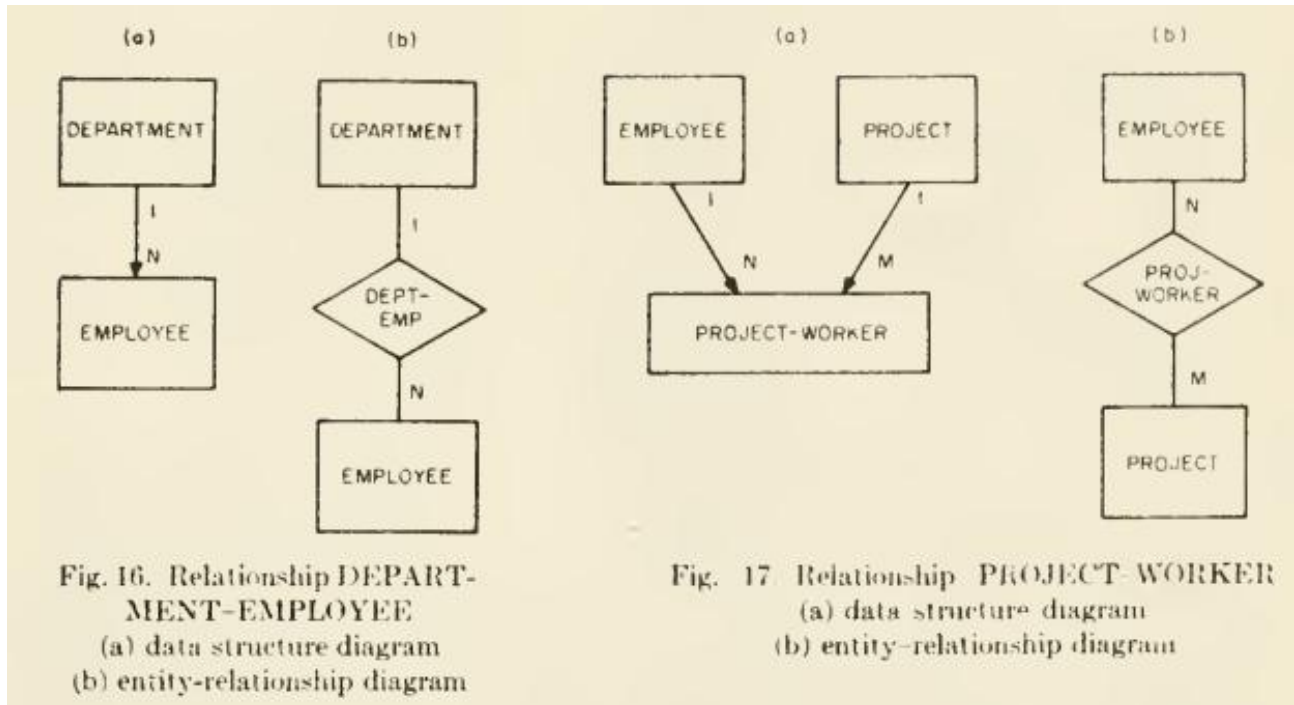
Note que en el ejemplo anterior, las relaciones entidad/relaciones son similares a las relaciones de la 3FN. En la cercana 3FN. PROJECT-MANAGER-NO es incluida en la relación PROJECT' desde PROJECT-MANAGER-NO se asume que es funcionalmente dependiente de PROJECT-NO. En el modelo entidad-relación, PROJECT-MANAGER-NO (EMPLOYEE-NO de un manejador de proyecto) se incluye en una relación de relación PROJECT-MANAGER desde EMPLOYEE-NO es considerada como una llave primaria de una entidad en este caso.

También note que en la próxima 3FN, los cambios en dependencias funcionales de datos pueden causar que algunas relaciones no estén en la 3FN. Por ejemplo, si hacemos una nueva suposición que un proyecto pueda tener más de un manejador, la relación PROJECT' ya no es una relación de la 3FN y tiene que ser separada en dos relaciones como PROJECT" y PROJECT-MANAGER". Usando el modelo entidad-relación no es necesario tal cambio. Por lo tanto podemos decir que usando el modelo entidad-relación podemos arreglar datos a una forma similar a relaciones de la 3FN pero con una clara definición semántica.

Es interesante notar que la descomposición (o transformación) aproximada descrita para normalización de relaciones puede ser vista como un enfoque de abajo hacia arriba en el diseño de bases de datos. Empieza con relaciones arbitrarias y después usa alguna información semántica (dependencias funcionales de datos) para transformarlos en relaciones de la 3FN. El modelo entidad-relación adopta un enfoque de arriba hacia abajo utilizando la información semántica para organizar datos en relaciones de entidad/relación.

## El modelo de red

Semántica del diagrama de estructura de datos. Una de las mejores maneras para explicar el modelo de red es usando un diagrama de estructuras de datos. La Figura 16(a) ilustra un diagrama de estructura de datos. Cada rectángulo representa un tipo de registro.



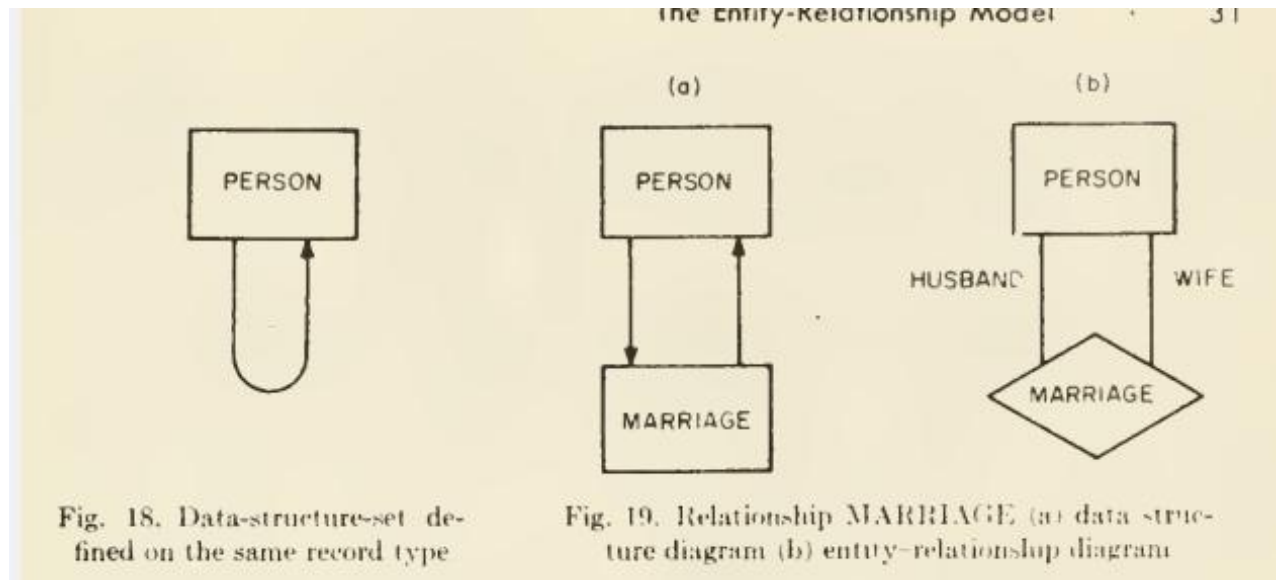
La flecha un conjunto de estructura de datos en donde el registro de DEPARTMENT es el registro del propietario, y un registro de propietario puede poseer  $n$  ( $n=0,1,2,\dots$ ) registros de miembros. La Figura 16(b) ilustra el diagrama entidad-relación correspondiente.

Uno puede concluir que la flecha en el diagrama de estructura de datos representa una relación entre entidades en dos conjuntos de entidad. Esto no siempre es cierto. Las Figuras 17(a) y 17(b) son los diagramas de estructura de datos y el diagrama entidad-relación expresando la relación PROJECT WORKER entre dos tipos de entidades EMPLOYEE y PROJECT. Podemos ver en la Figura 17(a) que la relación PROJECT-WORKER se convierte en otro tipo de registro y las flechas ya no representan relaciones entre entidades. ¿Cuáles son las definiciones reales de las flechas en diagramas de estructuras de datos? La respuesta es que una flecha representa una relación 1:N entre dos tipos registros (no es entidad) y también implica la existencia de una ruta de acceso del registro del propietario hacia los registros de miembros. El diagrama de estructura de datos es una representación de la organización de registros y no es una representación exacta de entidades y relaciones.

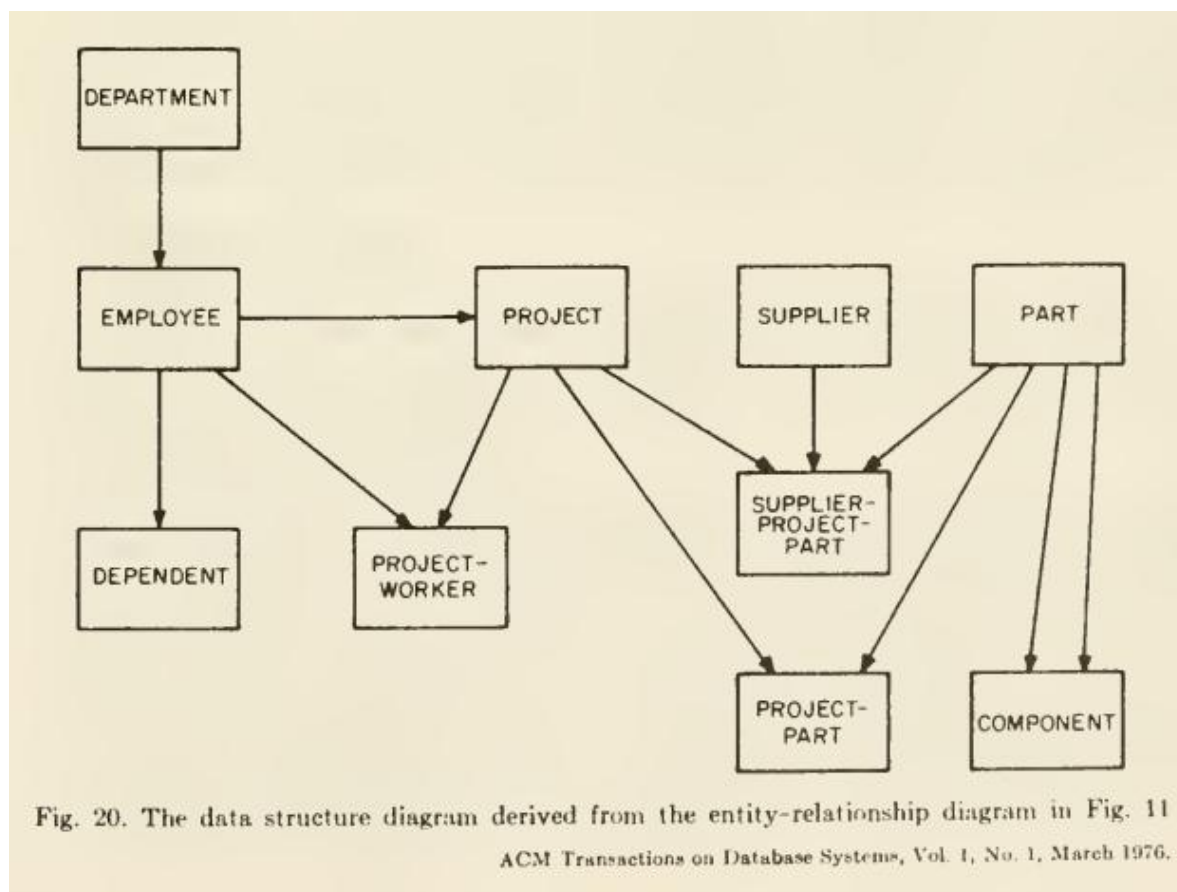
Derivando el diagrama de estructura de datos. ¿Bajo qué condiciones la flecha en un diagrama de estructura de datos corresponde a una relación de entidades? Una comparación cercana de los diagramas de estructura de datos con el diagrama entidad-relación correspondiente revela las siguientes reglas:

1. Para relaciones binarias 1:N una flecha es utilizada para representar la relación (ver Figura 16(a))
2. Para relaciones binarias M:N un tipo de “registro de relación” es creado para representar la relación y las flechas se dibujan desde el tipo de “registro de entidad” hacia el tipo de “registro de relación” (ver Figura 17(a))
3. Para K-ésimas relaciones ( $K \geq 3$ ), la misma regla (2) aplica.

Desde DBTG no se permite un conjunto de estructura de datos ser definido en un tipo de registro singular, un “registro de relación” necesita ser implementado como relaciones (ver Figura 19(a)). El diagrama entidad-relación correspondiente es mostrado en la figura 19(b).



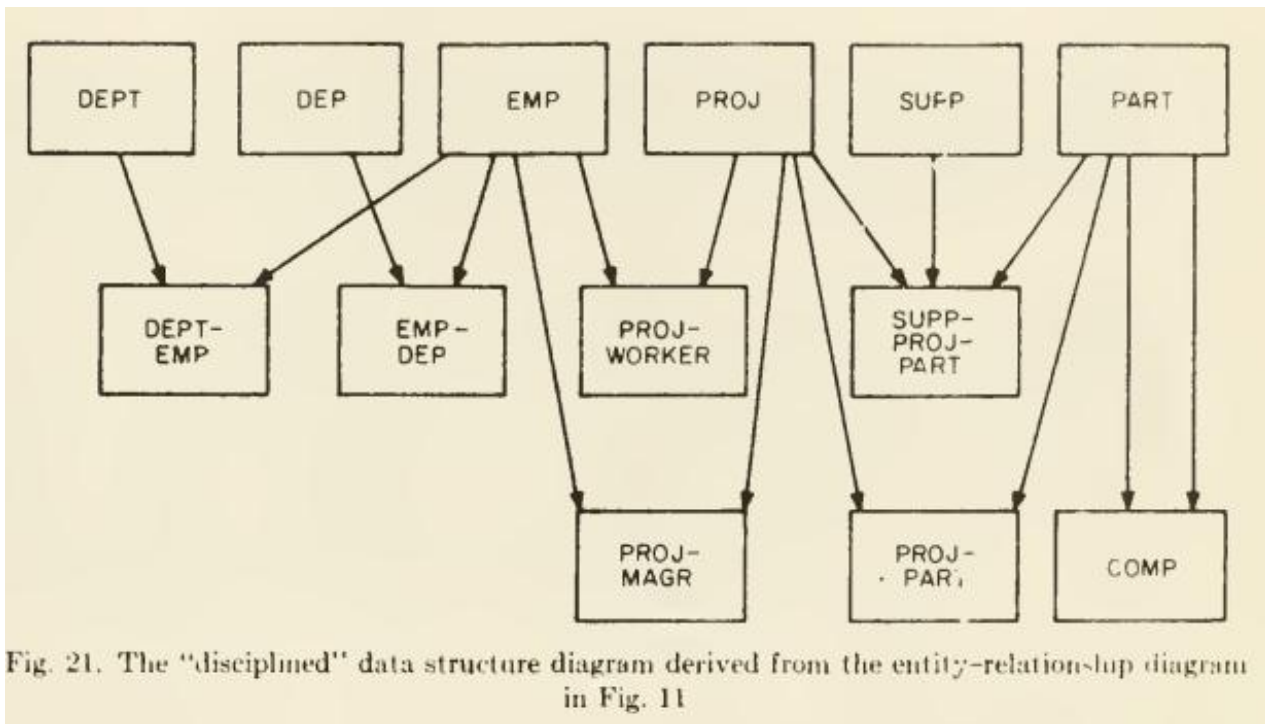
Ahora es claro que las flechas en un diagrama de estructura de datos no siempre representan relaciones de entidades. Incluso en el caso que una flecha represente una relación 1:N, la flecha solo representa una relación unidireccional, aunque es posible encontrar el registro de propietario de un registro de miembro. En el modelo entidad-relación, ambas direcciones de la relación son representadas (los roles de ambas entidades son especificadas). Además de la ambigüedad semántica en sus flechas, el modelo de red es incómodo en manejar cambios en la semántica. Por ejemplo, si la relación entre DEPARTMENT y EMPLOYEE cambia de una cardinalidad 1:N a una M:N, debemos crear un registro de relación DEPARTMENT-EMPLOYEE en el modelo de red.



En el modelo entidad-relación, todos los tipos de cardinalidades en las relaciones son manejadas uniformemente.

El modelo entidad-relación puede ser usado como herramienta en el diseño estructurado de las bases de datos usando el modelo de red. El usuario dibuja primero un diagrama entidad-relación (Figura 11). El simplemente lo traslada a en un diagrama de estructura de datos (Figura 20) usando las reglas especificadas. Él puede encontrar una disciplina que cada entidad o relación debe ser cardinalizada en un registro (esto es, "registros de relaciones" son creados para todos los tipos de relaciones no importa que tengan cardinalidades 1:N o M:N). En la Figura 11, todo lo que uno necesita hacer es cambiar los diamantes por rectángulos y añadir las cabezas de flecha en las líneas apropiadas. Usando la aproximación de 3 rectángulos más- DEPARTMENT-EMPLOYEE, EMPLOYEE-DEPENDENT y PROJECT MANAGER serán añadidos a la Figura 20 (ver Figura 21). Las restricciones de validación serán útiles.





### El modelo de conjunto de entidades

La vista del conjunto de entidad. El elemento básico del modelo de conjunto de entidad es la entidad. Las entidades tienen nombres (nombres de entidad) como "Peter Jones", "azul" o "22". Los nombres de entidad tienen algunas propiedades en común son recolectadas en un conjunto de nombres de entidad, que esta referenciado por el nombre de nombre de entidad como "NAME", "COLOR" y "QUANTIFY".

Una entidad es representada por el conjunto de nombre de entidad / el par entidad-nombre como NOMBRE/Peter Jones, EMPLOYEE-NO/2566, y NO-OF-YEARS/20. Una entidad es descrita por sus asociaciones con otras entidades. La Figura 22 ilustra el conjunto de vista de datos. El "DEPARTMENT" de la entidad "EMPLOYEE-NO/2566 es la entidad DEPARTMENT NO/405. En otras palabras, "DEPARTMENT" es el rol que la entidad DEPARTMENT-NO/405 trata de describir la entidad EMPLOYEE-NO/2566. Similarmente, el "NAME", "ALTERNATIVE-NAME", o "AGE" de EMPLOYEE-NO/2566 es "NAME/Peter Jones", "NAME/Sam Jones", o "NO-OF-YEARS/20", respectivamente. La descripción de la entidad EMPLOYEE-NO/2566 es una colección de entidades relacionadas y sus roles (las entidades y roles encerradas por la línea punteada). Un ejemplo de de la descripción de entidad de "EMPLOYEE-NO/2566" es ilustrada por el conjunto de roles de nombre, conjunto nombre de entidad/ conjunto de nombre de entidad, los triplete mostrados en la Figura 23, Conceptualmente el modelo de conjunto de entidad difiere del modelo entidad-relación en las siguientes maneras:

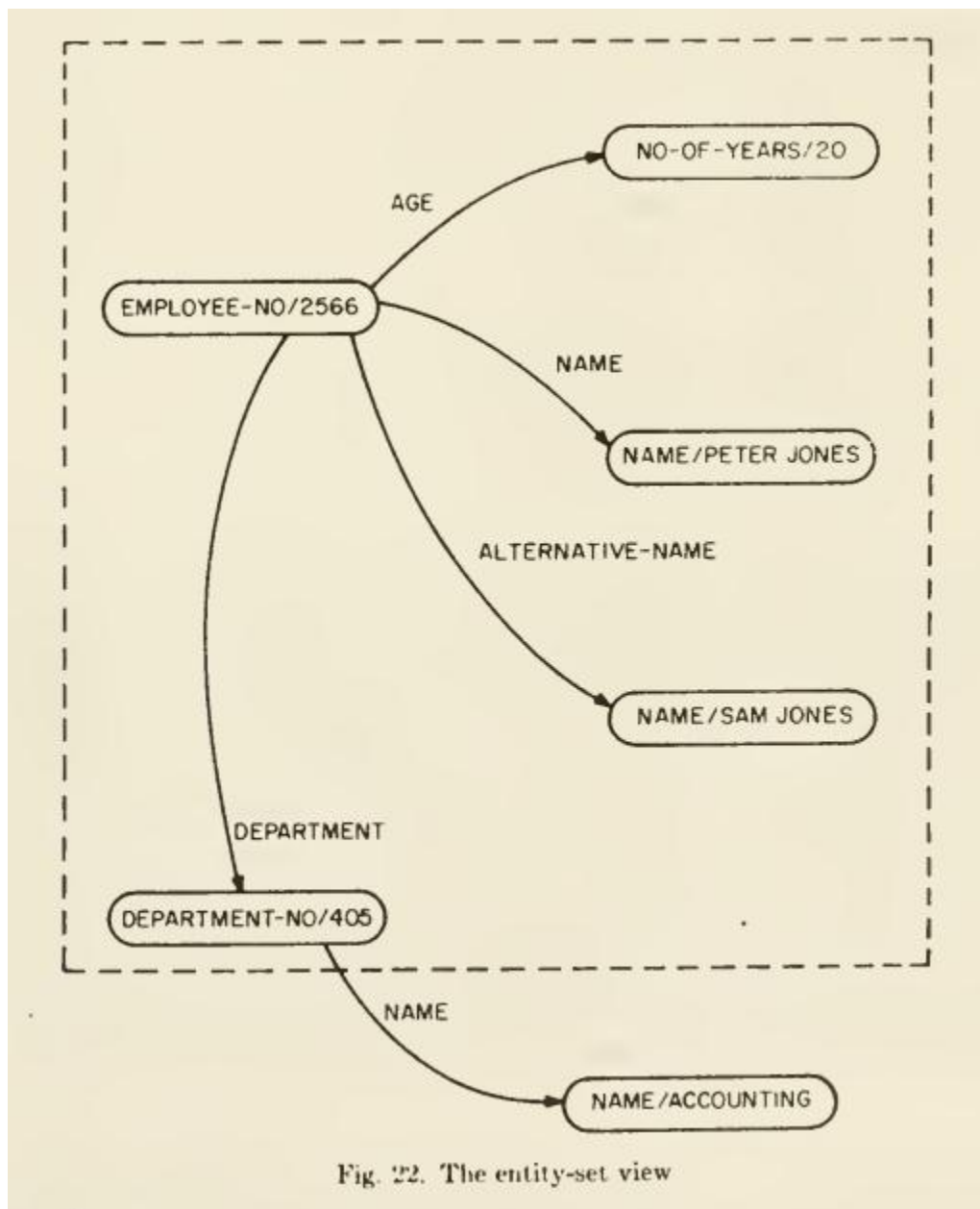
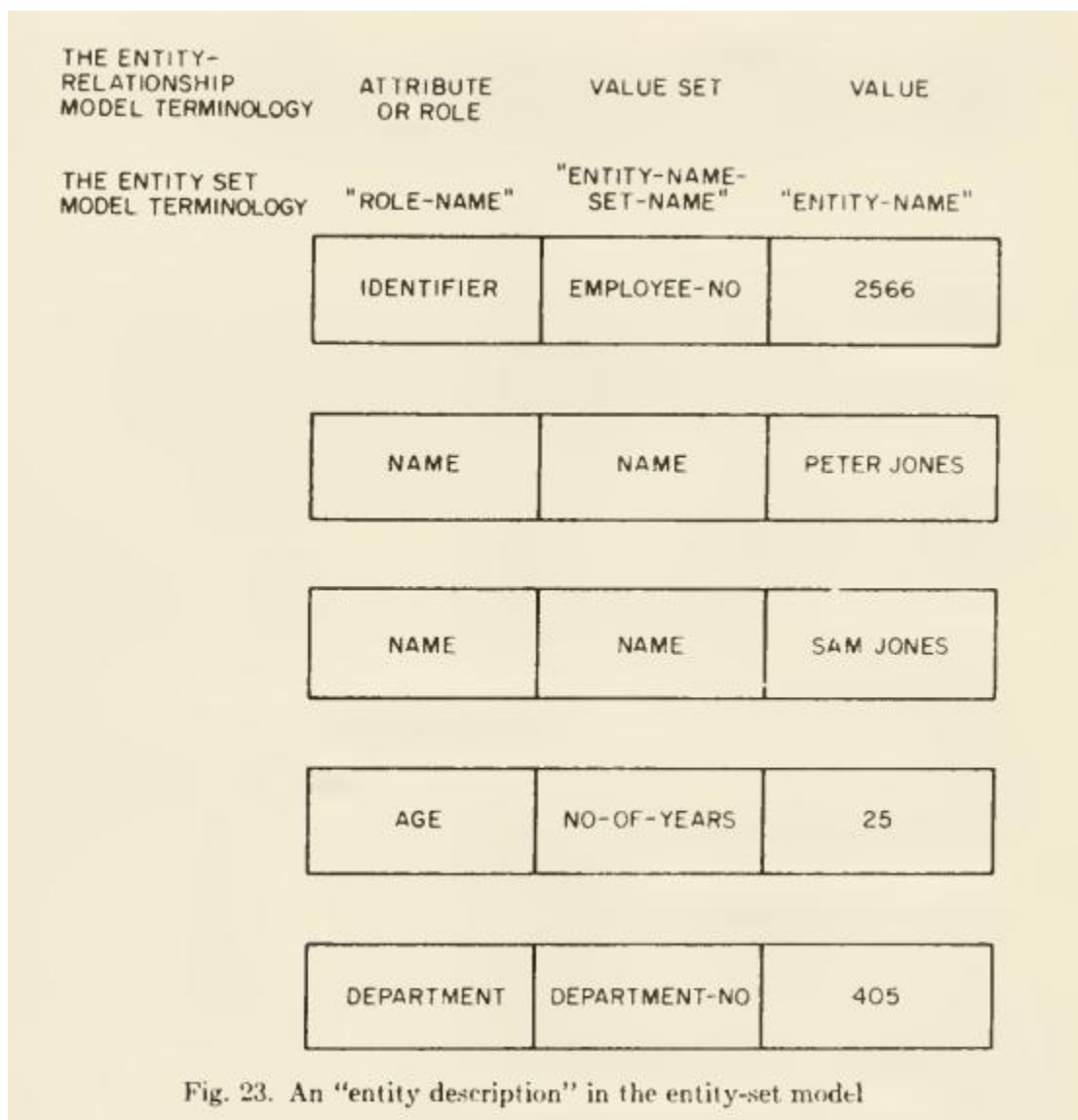


Fig. 22. The entity-set view

- (1) En el modelo de conjunto de entidad, todo es tratado como una entidad. Por ejemplo, "COLOR/BLACK" y "NO-OF-YEARS/45" son entidades. En el modelo entidad-relación, "blue" y "36" son usualmente tratados como valores. Note que tratar valores como entidades puede causar problemas de semántica. Por ejemplo en la Figura 22, ¿Cuál es la diferencia entre "EMPLOYEE-NO/2566", "NAME/Peter Jones" y "NAME/Sam Jones"? ¿Representan entidades diferentes?
- (2) Solo relaciones binarias son utilizadas en el modelo de conjunto de entidad, mientras relaciones n-ésimas pueden ser usadas en el modelo entidad-relación.



Derivando la vista del conjunto de entidad. Una de las principales dificultades en entender el modelo de conjunto de entidad, es debido a su cosmovisión (identificando valores con entidades). El modelo entidad-relación propuesto es útil para entender y derivar la vista de datos del conjunto de entidad.

En la Figura 6, las entidades son representadas por valores. El modelo de conjunto de entidad trabaja para el nivel 1 y el nivel 2, pero podemos explicar su vista en el nivel 2 (Figura 6). El modelo de conjunto de entidad trata todos los conjuntos de valor como NO-OF-YEARS como "conjuntos de entidad-nombre" y todos los valores como "nombres de entidad". Los atributos se hacen nombres de rol en el modelo de conjunto de entidad. Para relaciones binaria, la conversión es simple: el rol de una entidad en una relación (por ejemplo, el rol de "DEPARTMENT" en la relación DEPARTMENT-EMPLOYEE) se hace el nombre de rol de la entidad que es descrita en la otra entidad en la relación (Ver Figura 22). Para relaciones n-ésimas ( $n > 2$ ) debemos crear entidades artificiales para relaciones en el orden para manejarlas en un mundo de relaciones binarias.