WUOLAH

TEAM_GETPPID__
www.wuolah.com/student/TEAM_GETPPID__

37494

Bases de datos (Temario Resumido).pdf

Temario Resumido

2° Bases de Datos

Grado en Ingeniería Informática

Escuela Politécnica Superior de Córdoba
UCO - Universidad de Córdoba

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

Ismael Abu-Jadur Garcia

2017/2018

Bases de datos

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

Contenido

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

Tema 1: Introduccion a las Bases de Datos

1.1 Introducción

- Una base de datos o banco de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso.
- Inconvenientes pasados:
    - Hacían uso de ficheros planos y en ellas había gran redundancia y poca garantía en la integridad de los datos.
    - La estructura de archivos era secuencial (Se almacenaban de 1 en 1), siendo esto ineficiente, por ejemplo tardariamos muchos en la busqueda de un cliente.
    - Si se modificaban los campos o datos era casi imposible actualizarlos.
    - Si variaba el hardware el software debía de variar tambien.
- Con los dispositivos de almacenamiento que permitian el acceso directo los sistemas se hicieron más indepentendientes del hardware.
- IBM propuso la idea impulsada por JFK, en el proyecto apolo ya que necesitaban de un software capaz de almacenar a todos los trabajadores, componentes, fabricantes... Para tener un mejor control.
- IBM desarrolla las BDD jerarquicas con el software IMS. Una BDD jerarquica consiste en:
    - Toda la visión conceptual y física esta representada con forma de jerarquía donde el inicio es el root y cada uno de los problemas se denominan nodos. Cada nodo se forma de registros (Ocupan todos el mismo espacio) y cada registro por campos. Los nodos que dependen de otro son los hijos, siendo el nodo del que dependen el padre.
    - Las relaciones entre nodos se denominan links.
    - Un padre puede tener muchos hijos, pero cada hijo solo un padre, por lo que no podian existir las relaciones N:N a no ser que existiera una alta redundancia, por ello se plantea hacer una copia con "punteros" al original.
    - Esta estructura varía surgiendo las BDD en red, junto con los DBMS con el estandar codasyl, que nos permite conectar cada nodo como queramos con los demás.
    - Cada relación se llamaba de tipo set.
- Una base de datos garantiza la independencia de los datos respecto a los procedimientos. La independencia debe de satisfacerse a dos niveles de abstracción para que sea efectiva:
    - Independencia lógica: La modificación de la representación lógica general del dominio del problema no afecta a los programas de aplicación que la manipulan.
    - Independencia física: La distribución de los datos en las unidades de almacenamiento es independiente a la estructura lógica y, por tanto, de los procedimientos que manejan la misma.

1.2 Características de las bases de datos

- **Versatilidad para la representación de la información**: Pueden existir varias visiones de una misma información. Visiones parciales y/o visiones globales.
- **Desempeño**: Las bases de datos deben de asegurar un tiempo de respuesta adecuado, permitiendo el acceso simultaneo a los ítems de datos por el mismo o distinto procedimiento.
- **Capacidad de acceso**: Los usuarios de la base de datos reclaman a esta continuamente información sobre los datos. Una BDD debe de ser capaz de responder en un tiempo aceptable a cualquier consulta sobre la información que mantiene. Esta característica depende de la organización física de los datos en la BDD. También es necesario que cumpla al menos: Seguridad contra la destrucción de datos por el entorno, por causas del sistema, seguridad contra los accesos no autorizados o indebidos a los datos.

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

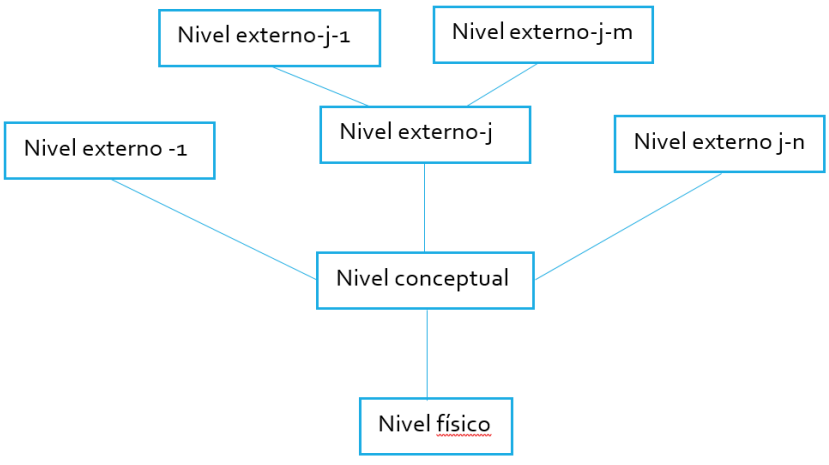
WUOLAH

- **Mínima redundancia**: Una de las principales razones por las que surgió la tecnología de las bases de datos fue el evitar la redundancia de las estructuras planas. Sin embargo, las BDD no evitan totalmente la redundancia debido a que es necesario representar todas las relaciones que existen entre las entidades que forman parte del dominio del problema. La existencia de redundancia es nefasta debido a la posibilidad de inconsistencia.
- **Simplicidad**: Deben estar basadas en representaciones lógicas simples que permitan la verificación, en la representación del problema que representan de tal forma que la inclusión y/o modificación de nuevos ítems de datos y relaciones no ocasione una complejidad excesiva.
- **Integridad**: Hace referencia a la veracidad de los datos almacenados con respecto a la información existente en el dominio del problema que trata la misma. Es necesario garantizar que estos datos no sean destruidos ni modificados de forma anómala, y en caso de que sucedan garantizar la integridad de la información.
- **Seguridad y privacidad**: Proteger los datos contra su pérdida total o parcial.
- **Afinación**: Organización física de la información de la base de datos, la cual determina directamente el tiempo de respuesta sobre los procedimientos que operan sobre la misma. Con el tiempo la envergadura de la información crece, por lo que la BDD debe de ser flexible a la modificación de la organización de la misma.
- **Interfaz con el pasado y el futuro**: El problema cambia, evolucionando con el tiempo. Una BDD debe de estar abierta a cambios de forma que no afecten a los procedimientos existentes para manejar la información que contiene. Además, debe de estar abierta a reconocer información organizada físicamente por otro software.

1.3 Visiones de los datos en las bases de datos.

- Para satisfacer las características señaladas es necesario tener una visión abstracta de los datos almacenados. Este modelo de BDD nos permite modificar cualquiera de las visiones sin depender de las demás.
- No es necesario conocer cómo se organizan los datos físicamente en la base de datos.
- Existen 3 visiones de los datos:
    - **Externa**: Es la visión que tienen los usuarios finales de una BDD. Un usuario trata sólo una visión parcial de la información, sólo aquella que interviene en el dominio de actividad. Estas **visiones particulares** de los usuarios son proporcionadas por los procedimientos o programas de aplicación que sólo manejan parte de la información de la base de datos.
    - **Conceptual**: Es la representación del problema tal y como este se presenta en el mundo real. Es una representación abstracta del problema e independiente, en principio, de cómo va a ser tratada esta información, de qué visiones externas pueda tener y de cómo esta información pueda ser almacenada físicamente

    - **Física**: Es la representación de cómo la información es almacenada en los dispositivos de almacenamiento, describe las estructuras u organizaciones físicas, dispositivos, volúmenes, ficheros, tipos de datos...



Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

La descripción de los dalos a estos tres niveles de abstracción diferentes garantiza la independencia de los datos, uno de los objetivos principales de las bases de datos:

- Que pueda ser modificada la organización de los datos sin que por ello haya cambiado o deba cambiar la descripción conceptual, y sin que por ello tengan que ser modificados los programas de aplicación.
- Que pueda ser modificada la representación conceptual del problema que está siendo representado en la base de datos sin que por ello tenga que ser modificada la estructura física de la información ni los programas de aplicación.
- Las visiones externas pueden variar conforme nuevos requisitos o necesidades funcionales, sin que por ello se deban modificar los niveles de abstracción.

Podemos conseguir una buena independencia de los datos si:

- La representación interna de los datos no es una traducción dependiente de la representación conceptual.
- Si las representaciones externas son dependientes de la representación, la estructura de los registros de los datos existentes debe de ser independiente a como los ítems han sido representados en el nivel conceptual.

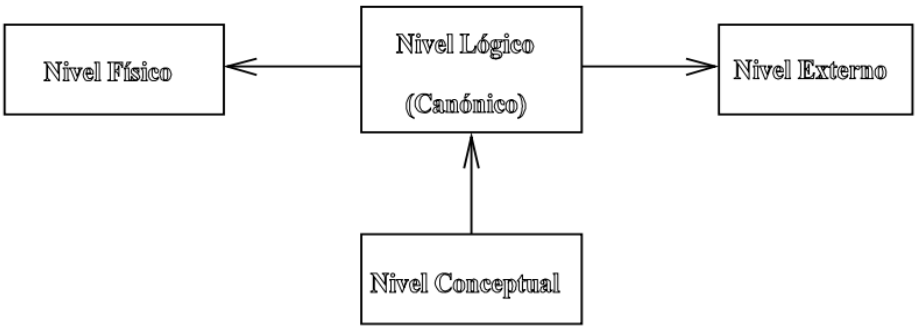
Independencia del nivel de descripción conceptual

- El nivel de descripción conceptual es el más importante, es aquel en el que se apoyan en menor o mayor grado los otros niveles.

- En este nivel se describe cada uno de los ítems de datos o elementos de información que intervienen en el comportamiento del sistema y cuya información es necesario considerar.

- Por ello, se puede hablar de un cuarto nivel de abstracción en la representación de la información en una base de datos, el nivel lógico o canónico, es independiente de las descripciones externas e interna. Si el problema no cambia, no cambia la representación conceptual, aunque cambien los mecanismos por los cuales el problema será tratado y, por tanto, alguna de las otras representaciones.

- La descripción conceptual depende del problema del mundo real objeto de la representación. Si el problema no varía, no varía la representación conceptual, aunque cambien los mecanismos por los cuales el problema será tratado (Nivel lógico).



1.4 Granularidad y ligadura

Granularidad: Es el nivel de detalle en que pueden ser descritas las representaciones externas derivadas de la representación lógica.

- Cuanta más granularidad de una representación externa (Menor información a considerar), mayor será la independencia y viceversa.
- Una mayor granularidad proporciona una mayor complejidad en el software utilizado para realizar estas representaciones.

Ligadura: La integridad de la BDD necesita que las representaciones (Procedimientos que la manejan) a un determinado nivel de abstracción tengan en cuenta como se representa la información en otros niveles. Las diferentes representaciones de los datos se vinculan entre sí (Ligadura).

- Existen 2 tipos:

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

- **Ligadura lógica**: Correspondiente al proceso de vinculación que se produce entre las representaciones externas y la lógica.
- **Ligadura física**: Correspondiente al proceso de vinculación entre la representación lógica y la física.

El proceso de vinculación entre las distintas representaciones puede realizarse en cualquiera de las siguientes fases: Compilación, enganche, ejecución y acceso a la base de datos.

Si la ligadura se realiza en una fase muy temprana (compilación, enganche) implica que los programas de aplicación deberán ser recompilados cada vez que se produzca una modificación de las distintas representaciones de los datos (lógica y física), aunque por otro· lado el desempeño de los mismos será alto debido a que el tiempo de cómputo que conlleva el proceso de ligadura se consume sólo una vez.

Por otro lado; el que se realice la ligadura en una fase tardía (en cada acceso a los datos) supone que se podrán modificar las representaciones lógica y física de los datos sin que por ello deban traducirse de nuevo a código máquina los programas de aplicación. Sin embargo, el desempeño de estos programas será en principio menor debido a que necesitan un coste computacional añadido para realizar el proceso de ligadura.

La solución de compromiso adoptada por muchos sistemas es que el proceso de ligadura se realice en la fase de ejecución.

1.5 Bases de datos y sistemas de gestión de BDD

Definimos una BDD: Es una colección de archivos relacionados que almacenan:

- Representación abstracta del dominio de un problema del mundo real
- Datos correspondientes a la información acerca del mismo
- Restricciones:
    - Innatas al problema
    - Garantizan la integridad

Definición completa de base de datos:

> *Una Base de Datos es una colección de archivos relacionados que almacenan tanto una representación abstracta del dominio de un problema del mundo real cuyo manejo resulta de interés para una organización, como los datos correspondientes a la información acerca del mismo. Tanto la representación como los datos están sujetos a una serie de restricciones, las cuales forman parte del dominio del problema y cuya descripción está también almacenada en esos ficheros.*

De esta podemos extraer:

- Una BD es una colección de archivos relacionados. Puede ser vista como un único depósito en el cual se almacena toda la información correspondiente al dominio de un problema.
- En estos archivos se encuentra la visión física, lógica y cada una de las visiones externas de la información, como los datos conocidos acerca del mismo en un momento dado.
- El que tanto la representación como los datos están sujetos a una serie de restricciones implica:
    - Las restricciones innatas al problema están representadas.
    - El acceso a la información almacenada está sujeto a una serie de restricciones que garantizan la integridad de la misma

1.6 SGBD

Un SGBD o sistema gestor de BDD, es una colección de programas que proporcionan al usuario de la BDD las herramientas necesarias para realizar las siguientes tareas:

- Definición de los datos a los distintos niveles de abstracción
- Manipulación de los datos en la BDD.
- Mantenimientos de la integridad de la BDD (Datos, valores y relaciones entre ellos).
- Control de privacidad y seguridad de los datos

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

- Medios necesarios para el establecimiento de todas aquellas características exigibles en una BDD.

Componentes de los SGBD

1. El lenguaje de definición de datos.

- El lenguaje de definición de datos (DDL) es un lenguaje artificial simple basado en un determinado modelo de datos que permite la representación lógica de los datos garantiza la definición no ambigüa de los datos.
- La representación de los datos obtenida en este proceso de compilación es almacenada en otro componente del SGBD denominado **Diccionario de datos**.

2. El lenguaje de definición del almacenamiento de los datos.

- El mismo lenguaje DDL permite la definición de los datos en el nivel de representación físico, si bien en otro es un subcomponente de este denominado DSDL (Data Storage Definition Language). En él se definen los datos correspondientes al dominio de un problema a los dos niveles de abstracción, y a esta definición de los datos se le denomina Esquema de la Base de Datos. En el esquema estarán definidas:
    - Las características del problema a un nivel de descripción lógico. Esta definición no variará a no ser que cambie el problema:
        - Cada una de las clases de objetos, y sus propiedades, que formen parte del dominio del problema.
        - Cada una de las relaciones, y sus propiedades, existentes entre estas clases de objetos.
        - Restricciones concernientes tanto a las clases de objetos y sus propiedades como a las relaciones entre ellos.
    - Características del problema desde un punto de vista físico:
        - Las unidades físicas en las cuales los datos van a ser almacenados
        - Los volúmenes y archivos utilizados
        - Las características físicas y lógicas de los medios de almacenamiento y
        - métodos de acceso a la información
- El DDL cuenta con otro sublenguaje encargado del control y seguridad, denominado DCL, y permite el control de acceso a la información almacenada en el diccionario de datos.

3. El lenguaje de manipulación de datos

- El lenguaje de manipulación de datos (DML) es un lenguaje que realiza dos funciones diferentes en la gestión de los datos:
    - Definición del nivel externo
    - Manipulación de los datos
- Dependiendo del modelo de datos en el cual se soportan y del SGBD, existen dos tipos:
    - **Procedimentales**: Requieren que en las sentencias del lenguaje se especifique acciones/operaciones a realizar.
    - **No procedimentales**: Los cuales el propio DML se encarga de determinar el procedimiento más efectivo.
- Mediante el DML se definen las vistas o visiones parciales que los usuarios tienen del esquema de la base de datosdefinido mediante el DDL. Estas vistas de los datos son denominadas Subesquemas y pueden realizarse de varias formas:
    - Haciendo uso única y exclusivamente del DML. Así, con sentencias propias de este lenguaje se definen distintas visiones parciales.
    - Haciendo uso de un lenguaje huésped (host) como C, COBOL, FORTRAN, etc., mediante el cual se realizan los programas de aplicación que permiten al usuario manipular los datos de la base de datos.

4. Diccionario de datos

- El diccionario de datos es uno o un conjunto de archivos que contienen información acerca de los datos que pueden ser almacenados en la base de datos.
- Se trata de una metabase de datos, una BDD que contiene información sobre otra BDD.
- En el Diccionario de datos se encuentra almacenado:
    - El esquema lógico de la BDD.
    - El esquema físico de la BDD
    - Subesquemas de la BDD

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

    - Restricciones de privacidad y acceso a los datos almacenados en la BDD (DDL o DCL)
    - Otra información que permite garantizar la integridad de los datos almacenados en la BDD
- En el diccionario de datos, además de almacenarse la representación de los datos al nivel externo, lógico y físico, se almacena un conjunto de reglas que permite vincular los mismos datos desde un nivel de abstracción y representación a otro. A este conjunto de reglas se le denomina Mapa de reglas.

5. El monitor de la BDD

- El monitor o gestor de la BDD es responsable de:
    - Garantizar la privacidad de los datos
    - Garantizar la seguridad de los datos, realizando los procedimientos necesarios para que los datos puedan ser recuperados en caso de fallo.
    - Garantizar la integridad de los datos, gestionando que los datos satisfagan las restricciones definidas en el esquema.
    - Garantizar el acceso concurrente a la BDD. varios usuarios puedan acceder al mismo o distinto dato
    - Interaccionar con el S.O y con el gestor de archivos, de forma que los procedimientos DML puedan ser entendidos por el S.O. Para ello, el gestor de la BDD cuenta con un subcomponente denominado procesador de consultas.
- Es un componente software encargado de garantizar el correcto, seguro, íntegro y eficiente acceso y almacenamiento de los datos. Este componente es el encargado .de proporcionar una interfaz entre los datos almacenados y los programas de aplicación que los manejan.

6. El administrador de la BDD

- Se trata de un componente humano de suma importancia en el resultado que el uso de las bases de datos va a tener en la resolución de un determinado problema.
- Entre las tareas asignadas al DBA:
    - Definición del esquema canónico o lógico de la BDD, la codificación mediante sentencias del DDL del conjunto de definiciones que representan las características del problema
    - Definición del esquema físico de la BDD, la especificación de las estructuras de almacenamiento y los métodos de acceso a la información almacenada en los dispositivos físicos.
    - Definición de los subesquemas o visiones externas, aquellas vistas parciales de la base de datos que son almacenadas en el diccionario de datos son definidas por el DBA
    - Control de la privacidad de los datos. mediante la concesión de privilegios a usuarios o grupos de éstos para el acceso a la información almacenada en la base de datos.
    - Mantenimiento de los esquemas:
        - Introducir las modificaciones necesarias en el esquema lógico.
        - Introducir las modificaciones necesarias en la representación física de los datos, de forma que esta representación evolucione paralelamente a la extensión de la base de datos.
        - Introducir las modificaciones y nuevas definiciones de los subesquemas o visiones externas.
    - Especificación de los procedimientos necesarios para el mantenimiento de la seguridad de los datos almacenados en la BDD.

7. Usuarios de la BDD

Se ha considerado a los usuarios de las bases de datos como un componente más de los SGBD:

- **Usuarios terminales**: Aquellos usuarios que a través de programas de aplicación interaccionan con la BDD.
- **Usuarios técnicos**: Aquellos que desarrollan los programas de aplicación que van a ser utilizados por los usuarios terminales del a BDD.
- **Usuarios especializados**: Aquellos que usan el SGBD como una herramienta en el desarrollo de otros sistemas más o menos complejos.
- **Usuarios críticos**: Aquellos usuarios gerenciales o pertenecientes al staff de las empresas en las cuales se ha instalado la BDD, los cuales, en base a expectativas, realizan consultas no previstas sobre la información almacenada en la BDD.

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

Tema 2: Representación de los problemas del mundo real

El proceso de interpretación de un fenómeno consiste en la propuesta de las propiedades o parámetros que caracterizan a este proponiendo un modelo inicial que intenta representar al mismo.

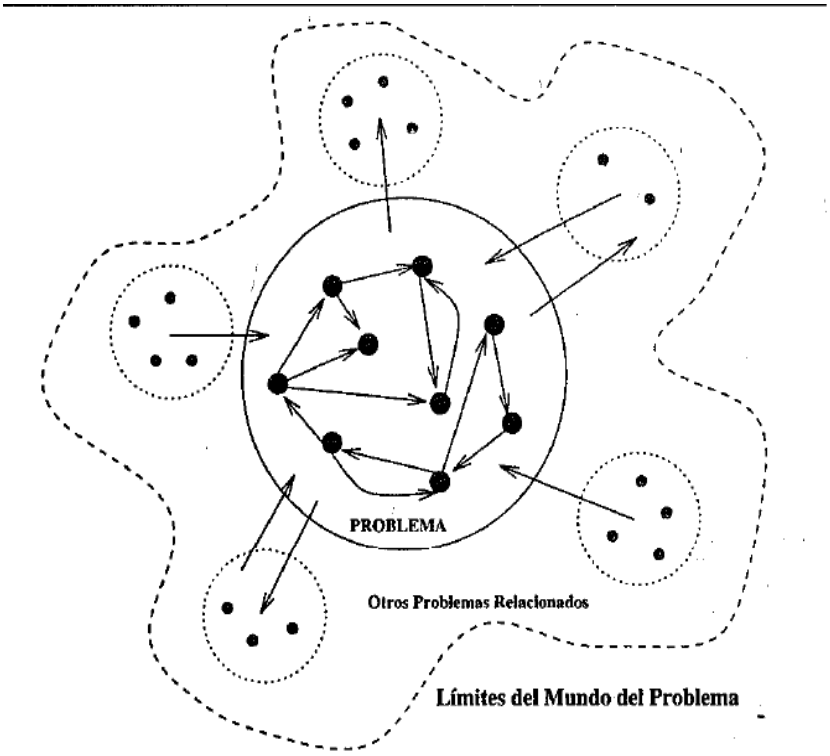
Al conjunto de las propiedades que caracterizan un fenómeno se le denomina datos al conjunto de valores que estas propiedades pueden presentar junto con el conjunto de las relaciones o dependencias se le denomina información.

Se definen unos tipos o clases abstractas de datos básicos los cuales pueden tomar un conjunto de valores predefinidos de antemano.

En definitiva, un modelo de datos describe las características estáticas y dinámicas de un fenómeno, es por tanto un conjunto de reglas que describen un fenómeno.

2.1 Los problemas del mundo real

El primer paso en la representación del conocimiento acerca de un problema del mundo real es la determinación de los límites del problema, determinando que datos y como pueden ser medidos.



Para el estudio de un sistema es necesaria la simplificación del problema que representa el mismo, que comienza por la determinación de los límites y determinando las propiedades de interés.

2.1.1 La abstracción

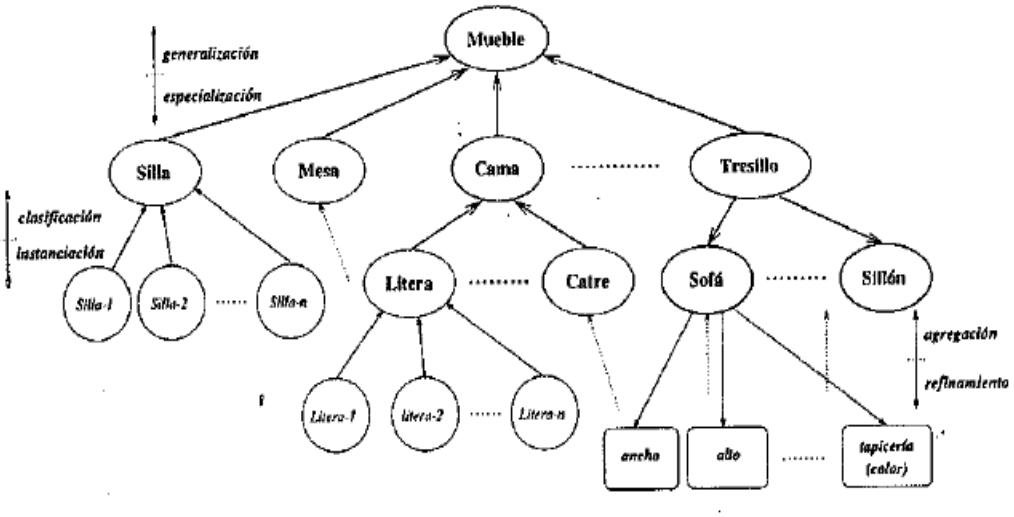
Capacidad mediante la cual una serie de objetos se categorizan en un nuevo objeto mediante una función de pertenencia, a este nuevo objeto se le denomina clase o tipo de objeto, todos estos objetos tienen propiedades comunes que caracterizan la clase.

La abstracción es utilizada de dos formas:

- Generalización: Un conjunto de clases de objetos puede ser visto como una nueva clase de objetos más general.
- Especialización: Proceso inverso a la generalización.

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH



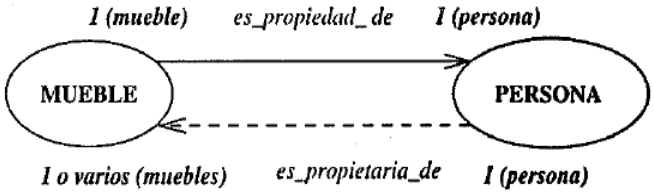
- Instanciación: Proceso mediante el cual creamos o declaramos objetos.
- Clasificación: El proceso inverso a la instanciación.
- Agregación: Es la capacidad de considerar un objeto basándose en los elementos que lo constituyen
- Refinamiento: Es el proceso contrario de la agregación, que es cuando definimos las características de un objeto.

2.1.2 Representación de los problemas del mundo real

Para la representación abstracta del problema, podemos comenzar de dos formas:

- Identificando los objetos más generales y procede un proceso de especificación e instanciación de los mismos.
- Determinando aquellos objetos más simples que intervienen en el problema y mediante un proceso de agregación y generalización llegar al objeto más general.

Además de las relaciones ES_UN y PARTE_DE consideradas entre las clases de objetos podemos establecer otras relaciones o dependencias entre elementos del mismo. Esta representación de otras relaciones debe consistir en la información correspondiente al significado de la dependencia y la información correspondiente al número de objetos que se ven implicados (Cardinalidad).



2.1.3 Análisis de los problemas

La representación de un sistema es la conclusión de un arduo y complejo proceso en el cual se determinan las entidades del sistema, sus dependencias y, por tanto, el comportamiento del mismo. Este proceso se denomina Análisis del Sistema.

La representación de un problema requiere el seguimiento de los siguientes pasos:

1. definición del problema, mediante una descripción simple y concreta del problema que se desea estudiar y de cuál es la función u objetivo que el sistema intenta alcanzar.
2. Definición de la arquitectura del problema, mediante una descripción de las partes importantes del sistema.
3. Definición de la estructura del problema, mediante la descripción de los elementos del sistema:
    a. Definición del objeto, mediante una descripción de la función que desempeña el objeto dentro del problema en estudio.
    b. Medida del objeto, mediante una descripción de los valores que pueden ser medidos o puede tomar el objeto para el problema en estudio.
    c. Relaciones entre los objetos, mediante una descripción de las interdependencias entre los objetos que intervienen en el problema.
    d. Restricciones inherentes a los objetos para el problema en estudio.

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

4. Definición de la dinámica del problema: Descripción de la evolución que el problema va a tener o tiene con el tiempo.
5. Estudio del comportamiento del modelo propuesto.

2.2 Modelos de Datos

Mediante un modelo de datos el sistema es descrito como una clase de objeto que interacciona con otras clases de objetos (otros sistemas), formado por otras clases, las cuales pueden a su vez ser clasificadas o refinadas. Además, se especifican las operaciones o acciones que los objetos pueden llevar a cabo. Un modelo de datos está a su vez formado por dos submodelos:

1. Un submodelo encargado de definir las **propiedades estáticas** del sistema.
2. Un submodelo encargado de describir las **propiedades dinámicas** del sistema.

De forma general todos los modelos presentan:

- Un **conjunto de reglas** mediante las cuales puede ser representado gráficamente el problema soportado por un conjunto de símbolos por los que pueden ser representados cada uno de los objetos del sistema a los diferentes niveles de abstracción y cada una de las relaciones o dependencias.
- Un **pseudolenguaje** formado por un conjunto reducido de morfemas y una sintaxis perfectamente establecida, mediante el cual pueden describirse las propiedades estáticas y dinámicas del sistema.
- Un **conjunto de restricciones** del modelo que marcan los límites de los sistemas a representar y, por tanto, de las características de los sistemas que pueden ser representados por un modelo.

Los niveles de abstracción (Conceptual, lógico y físico) ya descritos previamente podemos verlos en la siguiente tabla:

NIVEL DE DESCRIPCIÓN	ESTRUCTURA	COMPORTAMIENTO
Modelo Conceptual	Descripción de los objetos del mundo real, de sus atributos o propiedades y de las relaciones entre los objetos	Descripción del comportamiento de los objetos: las acciones, operaciones y procesos que estos objetos realizan sobre otros objetos, así como las que son realizadas sobre los objetos del sistema
Modelo Lógico **Modelo Organizativo**	Descripción de los objetos, así como las relaciones existentes entre los objetos lógicos, identificando los atributos por los cuales estos objetos pueden ser identificados	Descripción de las tareas que se deben realizar para representar el comportamiento de los objetos. Estas tareas se agruparán en fases y procedimientos
Modelo Físico **Modelo Procedimental**	Descripción de los objetos físicos. La estructura y relaciones de los objetos son representadas de forma adecuada para su posterior almacenamiento, recuperación y tratamiento	Descripción de las acciones elementales que se deben realizar para representar el comportamiento de los objetos. Estas acciones son representadas bajo las limitaciones del lenguaje que se vaya a utilizar para su implementación en lenguajes de ordenador

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

2.3 Modelo Entidad-Interrelación

Propuesto por Peter Chen a mediados de los años setenta para la representación conceptual de los problemas y como un medio para representar la visión de un sistema de forma global, tratándose de un modelo muy extendido que ha experimentado muchas ampliaciones originando un potente medio para la representación de datos.

El Modelo E-R propone el uso de tablas bidimensionales para la representación particular de cada uno, y por tanto, de los conjuntos de elementos particulares y sus relaciones.

Definiciones:

- **Conjunto**: Agregación de una serie de objetos elementales mediante una función de pertenencia.
- **Relación**: Es un nuevo conjunto en el que cada elemento está formado por la agregación de los elementos de los conjuntos individuales que intervienen en la misma. El orden de la relación es importante. Una relación puede ser binaria, ternarias...
- **Intención y extensión**: La **intención** representa la clasificación de una serie de elementos individuales en un tipo o clase de objeto al que se ha denominado conjunto o relación. La **extensión** representa la instanciación de un tipo o clase de objetos.
- **Dominio**: Aquellos conjuntos cuyos elementos son homogéneos. Podemos definirlo también como una especialización de un conjunto.
- **Atributo**: Denominado así a la intención de un dominio y el valor del atributo la extensión del dominio. Identifica la semántica de un dominio para la descripción de un problema.
- **Entidad**: Es un tipo de objeto (Un conjunto) definido en base a la agregación de una serie de atributos. Corresponde a la caracterización de objetos del mundo real diferenciados del resto por una serie de atributos. La intención de una entidad es denominada tipo de entidad que representa la clasificación de las entidades individuales.
- **Interrelación**: representa la relación existente entre entidades, denominándose tipo de interrelación a la intención de la relación existente entre dos tipos de entidad. Un conjunto de interrelaciones representa a cada una de las posibles correspondencias entre los conjuntos de entidades que intervienen en la interrelación.

2.3.1 Entidades e interrelaciones en el modelo E-R

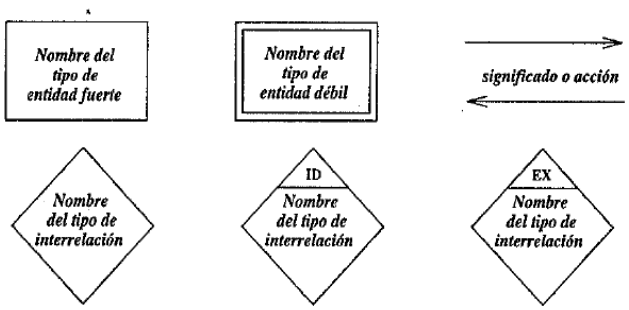
En el modelo E-R se considera que una entidad es un objeto real o abstracto que forma parte del sistema o problema en estudio y que cumple las siguientes propiedades:

- **Tiene existencia propia,** la entidad existe como un elemento que interviene en el comportamiento global del sistema.
- Es **distinguible** del resto de las entidades
- Las entidades de un mismo tipo están **definidas en base a un mismo conjunto de atributos**

Un tipo de interrelación es una relación matemática entre n tipos de entidades.

Consideramos dos tipos de entidades:

- **Fuertes**: Su existencia no depende de la existencia de ningún otro tipo de entidad del problema.
- **Débiles**: Su existencia depende de la existencia de un tipo de entidad fuerte. Estas las podemos clasificar en dos tipos:
    - **Débiles por identificación**: Una entidad débil no puede ser identificada, a no ser que se identifique mediante una entidad fuerte por cuya existencia está presente la debilidad.
    - **Débiles por existencia**: Esta entidad puede ser identificada sin necesidad de una fuerte, pero su existencia depende de ella.



Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

Una debilidad de identificación implica una debilidad de existencia, pero no al contrario.

Un tipo de interrelación fuerte representa la relación entre dos tipos de entidad fuertes y, viceversa, un tipo de interrelación débil representa la relación entre un tipo de entidad fuerte y una débil, o bien, dos tipos de entidad débiles.

Los **tipos de entidad son representados mediante un rectángulo** etiquetado, y **los tipos de interrelación mediante un rombo** igualmente etiquetado.

Los tipos de **entidad débiles** son representados con un **doble rectángulo etiquetado**, y **los tipos de interrelación débiles** mediante un **rombo en el cual se indica el tipo de debilidad** existente (**EX**istencia, **ID**entificación).

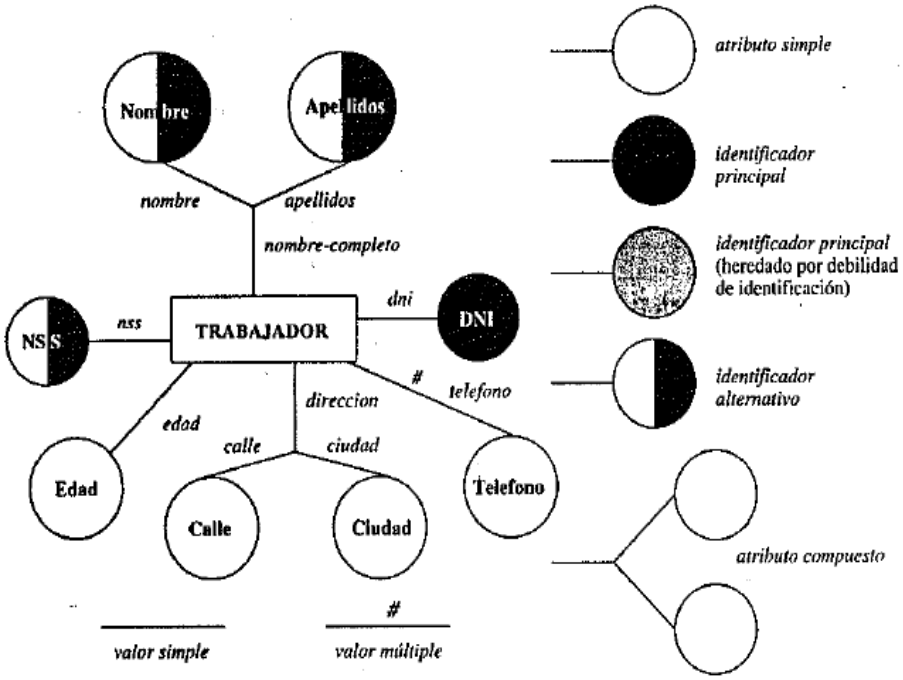
En los arcos que unen los tipos de interrelación con los tipos de entidad **se describe el significado en el mundo real**.

En un diagrama E-R es necesario representar, para cada tipo de interrelación, **las cardinalidades con las que cada tipo de entidad interviene en el tipo de interrelación**. Las cardinalidades se representan mediante una pareja de datos (en minúsculas) en la forma: (cardinalidad mínima, cardinalidad máxima).

Es conveniente también acompañar a la representación de los tipos de interrelación las **cardinalidades máximas** (en mayúsculas) con las que intervienen los tipos de entidad relacionados en el tipo de interrelación.

2.3.2 Descripción de los tipos de entidad e interrelación.

La representación de los atributos para un tipo de entidad es la siguiente:



Para eliminar cualquier ambigüedad en la representación de los objetos, tanto los tipos de entidad como los tipos de interrelación deben tener asignado un nombre único en el modelo.

Se denomina identificador de un tipo de entidad al conjunto de atributos (tal vez uno sólo) que no toma el mismo valor para dos entidades diferentes del mismo tipo. Para un tipo de entidad puede haber más de un conjunto de atributos que satisfagan esta condición, siendo candidatos para desempeñar este papel. Si no encontramos ningún identificador y no es una entidad débil, debemos de crear uno nuevo que satisfaga esta propiedad de unicidad.

Una interrelación es identificada, generalmente, por la concatenación o agregación de los atributos que identifican las entidades relacionadas.

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

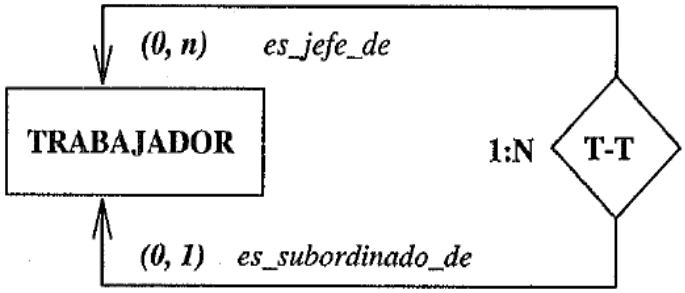
WUOLAH

2.3.3 Los tipos de interrelación en el modelo E-R

Las últimas actualizaciones del modelo E-R, que han dado lugar a lo que se denomina Modelo Entidad-Interrelación Extendido (EE-R), permiten la representación de cualquier tipo de relaciones existentes'. entre clases de objetos que considera los principios de la abstracción.

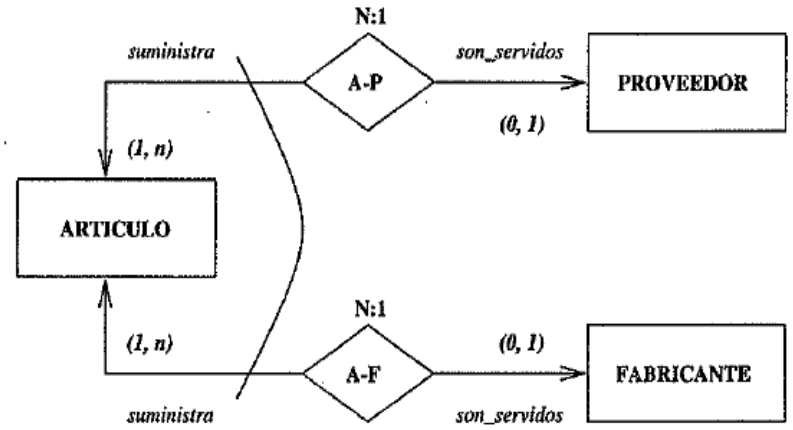
2.3.3.1 Interrelaciones reflexivas

Son relaciones unarias y se trata de un tipo de interrelación reflexiva en la que interviene un único tipo de entidad que desempeña dos papeles diferentes en el mismo tipo de interrelación.



2.3.3.2 Interrelaciones exclusivas

Para indicar la exclusividad entre dos tipos de interrelación que mantiene un mismo tipo de entidad se procede a representar un segmento que corta a los dos arcos que representan la relación del tipo de entidad con los tipos de interrelación exclusivas.



2.3.4 Generalización y herencia en el modelo E-R

El modelo EE-R permite representar las relaciones jerárquicas existentes entre los tipos de entidad de los problemas del mundo real. Este tipo de relaciones entre tipos de entidad implica la consideración de tipos de entidad (o supertipos) y de subtipos de entidad (clases, superclases y subclases de objetos).

Un subtipo de entidad es un tipo de entidad que mantiene un tipo de interrelación jerárquica con otro tipo de entidad, y que:

- Representa a un conjunto de entidades cuyas propiedades y comportamiento general es considerado por la entidad supertipo.
- La relación jerárquica puede ser n-aria entre un tipo de entidad y un conjunto de subtipos.
- Las propiedades y el comportamiento de los subtipos son heredados del supertipo.
- Las propiedades y/o bien el comportamiento de un subtipo pueden y deben cambiar con respecto a otros subtipos que intervengan en la misma relación. Estos deben poder distinguirse sin ambigüedad.
- Para cada subtipo de entidad pueden ser redefinidas tanto las propiedades como el comportamiento del supertipo.
- Un tipo de entidad puede ser un subtipo para más de un tipo de entidad con las que puede mantener diferentes relaciones jerárquicas.

Un tipo de interrelación jerárquica representa una especialización de un tipo de entidad en otros tipos de entidad. Esta especialización puede ser debida a:

1. Diferencia en cuanto al número de propiedades que definen los subtipos de entidad.
2. Diferentes valores que pueden ser medidos para una propiedad que existe en el conjunto de subtipos.
3. Que se cumplan ambas condiciones.

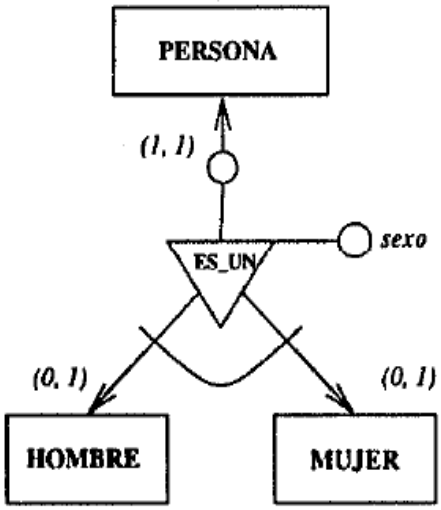
Esta especialización puede ser **exclusiva** que representa el hecho de que una instancia del supertipo solo pueda pertenecer o estar asociada a una instancia de los subtipos o **inclusiva** representa el hecho de que una instancia del tipo de entidad más general puede tener asociadas instancias de cualquiera de los subtipos.

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

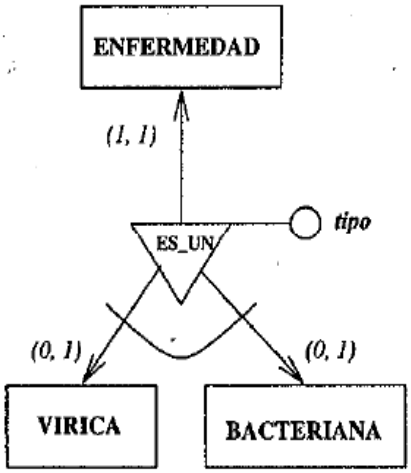
WUOLAH

Por otro lado, la especialización puede ser **total** que representa el hecho de que las entidades son de alguno de los subtipos especializados, no existiendo entidades que no pertenezcan a ningún subtipo o **parcial** representando el hecho de que pueden existir entidades que pertenezcan al tipo de entidad y no a ninguno de los subtipos especializados.

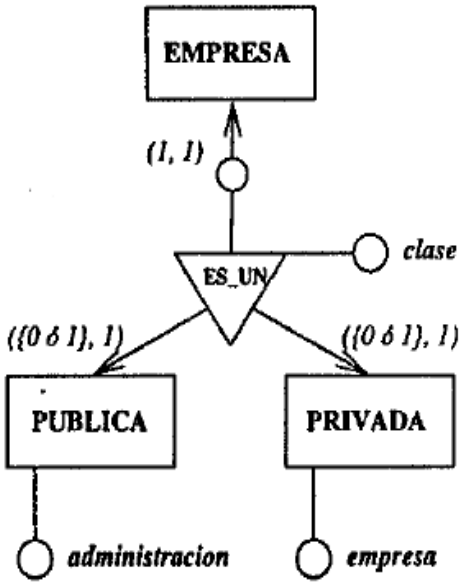
Total exclusiva:



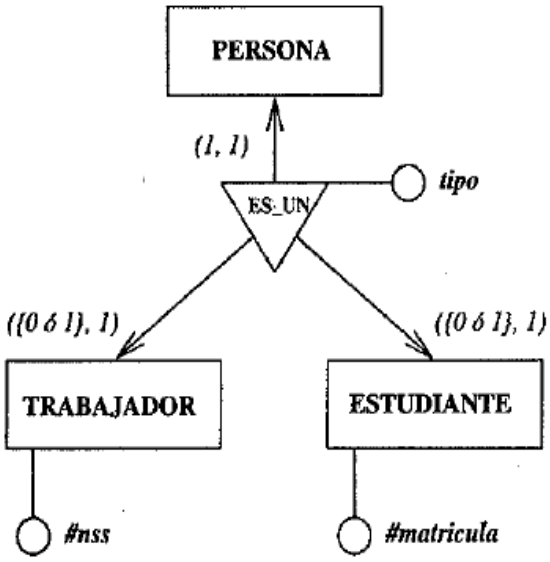
Parcial exclusiva:



Total inclusiva:



Parcial inclusiva:



2.3.4.1 Cardinalidades en la jerarquía

Dependiendo del tipo de interrelación jerárquica que se represente los tipos de entidades que intervienen participan con un N.º de ocurrencias. Por lo que tenemos que tener en cuenta:

- La entidad supertipo participa siempre con cardinalidad mínima 1 y con cardinalidad máxima 1, ya que representa una entidad que se puede especializar en otros.
- La entidad de los subtipos tiene una cardinalidad máxima de 1, ya que se representa una especialización.
- Si es total o parcial sin solapamiento los subtipos tendrán cardinalidad mínima 0, puesto que una entidad solo pertenece a un subtipo únicamente.
- Si la interrelación es total o parcial con solapamiento los subtipos tendrán cardinalidad mínima 0 ó 1, ya que se puede especializar en varios subtipos.

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

Tema 3: El modelo de datos relacional

3.1 La teoría relacional

No todos los modelos de datos dan lugar a esquemas complejos cuando el propio problema que representa es complejo o ha experimentado ampliaciones/modificaciones, ni tampoco una representación compleja es directamente causada por el uso de un determinado modelo de· datos. La complejidad del esquema va a depender tanto de:

- La complejidad del problema del mundo real
- La calidad del análisis de ese problema
- La evolución del problema y de los requisitos funcionales

Si bien es obvio que el uso de un modelo, que por su propia teoría en la que esté basado se vea menos influenciado por estos factores, dará lugar a una representación lógica de los problemas que origine esquemas que sean más entendibles, simples, claros y, por tanto, más fácilmente mantenibles, seguros y confiables que con el uso de otros modelos más sensibles a estos cambios. Y es ésta la base de la aparición del **Modelo de datos relacional**.

3.2 El modelo de datos relacional

Este modelo fue desarrollado por E.F. Codd, basado en conceptos sencillos y asociando la teoría de la normalización de relaciones que tiene por objeto la eliminación de los comportamientos anómalos de las relaciones, la eliminación de redundancias superfluas y facilitando la compresión del esquema.

El modelo relacional propone una representación de la información que:

- Origine esquemas que representen fielmente la información
- Pueda ser entendida fácilmente por los usuarios
- Haga posible ampliar el esquema de la base de datos sin modificar la estructura existente
- Permita la máxima flexibilidad en la formulación de los interrogantes previstos y no previstos

3.2.1 Terminología del modelo relacional

Para la representación de la información se hace uso de tablas bidimensionales la cual contendrá tanto como objetos, como las relaciones entre ellos.

Una tabla es una matriz rectangular, descrita matemáticamente y que posee las siguientes propiedades:

- Cada entrada de la tabla representa un ítem de datos elemental
- Una tabla es homogénea por columnas (Los datos de una misma columna son de la misma clase)
- Cada columna tiene asignado un nombre único en esa tabla.
- Para una tabla todas las filas son diferentes.
- Tanto las filas como las columnas pueden ser consideradas en cualquier orden.

El modelo relacional introduce su propia terminología para nominar los objetos y elementos.

- Una tabla se denomina ***relación***
- Cada fila es una ***tupla***
- Al conjunto de las columnas se le denomina ***dominio de la relación***

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

3.2.1.1 Relación

Dada una serie de conjuntos *D1, D2, D3,... Dn* no necesariamente distintos, se dice que R es una relación entre estos *n* conjuntos si es un conjunto de *n* tuplas no ordenadas (d1, d2. D3. ..., dn) tales que d1 ∈ D1, d2 ∈ D2, d3 ∈ D3, ..., dn ∈ Dn.

A los conjuntos *D1, D2, D3,... Dn* se les denomina dominios de R y el valor de *n* el grado de la relación.

- Al número de tuplas de una relación en un instante se le denomina cardinalidad de la relación.
- Al número de columnas de una relación se le denomina grado de la relación

3.2.1.2 Dominios y atributos

- Un atributo aporta un significado semántico a un dominio.
- Un dominio es un conjunto homogéneo definido mediante el uso de la abstracción

3.2.1.3 Intención y extensión de relaciones

Una relación en una base de datos relacional tiene dos componentes: la intención y la extensión.

- La intención de una relación hace referencia a la estructura estática del objeto del mundo real (invariante con el tiempo y se trata de la definición de las propiedades). La intención de una relación define dos aspectos:
    - Una estructura de datos nominada, en la que el objeto y los ítems que lo componen tienen un nombre único y están definidos en un dominio
    - Un conjunto de restricciones de integridad
- La extensión de una relación depende del momento, y hace referencia al conjunto de tuplas que forman parte de la relación en un instante dado

3.2.2 Consistencia de la representación lógica relacional

La representación lógica de un problema, independientemente del modelo utilizado, debe satisfacer la representación conceptual de ese problema y. por tanto, las especificaciones, restricciones y requisitos definidos en el proceso de análisis del problema del mundo real que se esté tratando.

3.2.2.1 Claves de las relaciones

A los atributos, tal vez compuestos. que satisfacen la propiedad de identificación única de las tuplas de una relación se les denomina *claves candidatas* de la relación. Toda relación, por definición, debe tener alguna clave candidata.

De entre todas las claves candidatas de una relación, en la definición del esquema se deberá especificar cuál de ellas se considera como *clave primaria o principal*, siendo el resto claves alternas.

La elección de los atributos (uno o varios) que forman parte de las claves candidatas no es un proceso trivial.

3.2.2.2 Integridad de los esquemas relacionales

La intención de un esquema relacional debe satisfacer las siguientes reglas de integridad mediante las cuales se garantiza la consistencia de la información que pueda ser manejada sobre la base de ese esquema:

- **Integridad de la clave**: Ningún atributo que forme parte de la clave candidata de una relación podrá tomar valores nulos para ninguna tupla de esa relación.
- **Integridad de referencia**: Sea D un dominio primario, y sea R1 una relación con un atributo "a" definido sobre el dominio D, entonces, en cualquier instante dado, cada valor de "a" en R1 debe ser nulo o bien igual a algún valor V, el cual existe, en ese instante. paro un atributo "b" definido en el mismo dominio D sobre la relación R: y en la cual está definido como clave primaria.

  A aquellos atributos que satisfacen esta regla de integridad se les denominan claves foráneas.

- **Otras restricciones**: En la definición del esquema relacional pueden imponerse, otra serie de restricciones que garanticen la integridad del modelo:
    - Los valores permitidos para los atributos que forman parte de las relaciones existentes en el esquema
    - Condiciones que determinan el valor que pueden tomar los atributos.

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

3.3 Normalización de relaciones

Si bien una relación representa a un conjunto, y como tal puede y debe ser considerada, no todos los conjuntos pueden ser considerados en un esquema relacional para satisfacer en la base de datos los siguientes objetivos:

- No-existencia de redundancias superfluas, aminorando el espacio requerido para el almacenamiento
- Aumentar el desempeño de las operaciones de actualización de la base de datos.
- Representar de forma coherente los objetos y relaciones existentes
- Aumentar el desempeño y garantizar la fiabilidad

Para satisfacer estos objetivos, las relaciones que forman parte de un esquema relacional deben satisfacer una serie de reglas. A este conjunto de reglas se les denomina Reglas de normalización de relaciones.

3.3.1 Dependencias funcionales

La normalización de relaciones está basada en la teoría de las dependencias, la cual se centra en el estudio de l□s dependenci.as que presenta cada atributo de una relación con respecto al resto de atributos de la misma relación.

Dada una relación R, se dice que el atributo "y" ∈ R es funcionalmente dependiente de otro atributo "x" ∈ R, y se expresa de la forma "x" → "y" si, y sólo si, cada valor de "x" tiene asociado a él exactamente un valor de "y" para cualquier extensión de la relación R.

De la definición anterior tenemos en cuenta que:

- La dependencia funcional tiene en cuenta a atributas de la misma relación y no de otras.
- Hace referencia a la relación funcional que pueda existir entre parejas de atributos.
- Una dependencia funcional es independiente del estado o extensión de una relación, siendo intrínseca a la intención de una relación.
- No hace referencia explícita a la complejidad de los atributos dependientes, por lo que:
    - Los atributos funcionalmente dependientes de una relación pueden ser simples o agregados.
    - Los atributos funcionalmente dependientes pueden ser, o no, atributos que formen parte de la clave primaria o clave candidata.
- En esta definición no se hace referencia al número de veces (N.º de tuplas) en las que el atributo "x" tienen un mismo valor.

En la definición no se impone la restricción de que no puedan repetirse los valores del atributo "y" para distintos valores del atributo "x".

Extendemos la definición a:

Dada una relación R, se dice que el atributo "y" ∈ R es funcionalmente dependiente de otro atributo "x" ∈ R, y se expresa en la forma "x" → "y" si, y sólo si siempre que dos o más tuplas de R coincidan en sus valores de "x", también, para esas tuplas, existirá una coincidencia en los valores del atributo "y".

Representación textual de las dependencias funcionales

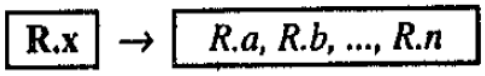
Formato general

$$R.x \rightarrow (R.a, R.b, \ldots R.n)$$

denotando que los atributos *R.a, R.b, ..., R.n* son funcionalmente dependientes del atributo *R.x.*

Representación gráfica de las dependencias funcionales

Formato general



Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

El concepto de dependencia funcional completa se define:

Se dice que el atributo "y" ∈ R es funcionalmente dependiente y de forma completa de otro atributo "x" ∈ R si, y sólo si, depende funcionalmente de "x" y no de ningún subconjunto de los atributos que formen parte del atributo "x"

Apreciando entonces:

- Si el atributo "x" es un agregado formado por varios atributos pertenecientes a la relación, entonces la dependencia funcional "*x*" → "*y*" no será completa.
- Que el atributo "y" sea simple o compuesto no tiene relevancia para la dependencia funcional.

3.3.1.1 Propiedades de las dependencias funcionales

- **Reflexiva**: Dados los atributos "a" y "b" ∈ R, para los que se cumple que "b" ⊆ "a" entonces, en la relación R, está presente una dependencia funcional de forma que "*a*" → "*b*".

- **Aumento**: Dados los atributos a y b de una relación R en la que está presente la dependencia funcional "*a*" → "*b*" entonces también estará presente la dependencia funcional (a+c) → (b+c), siendo "c" cualquier otro atributo que forme parte de la intención de la relación R.

- **Transitiva**: Dados los atributos "a", "b" y "c" de una relación R en la que están presentes las dependencias funcionales "a" → "b" y "b" → "c", entonces también estará presente la dependencia funcional "a" → "c"

A estas tres reglas descritas se les denomina Axiomas de Armstrong y de ellos deducimos otro conjunto de propiedades:

- **Unión**: Dados los atributos "a", "b" y "c" de una relación R en la que están presentes las dependencias funcionales
- "a" → "b" y "a" → "c", entonces también estará presente la dependencia funcional "a" → ("b" + "c")

- **Pseudo-transitiva**: Dados los atributos "a", "b", "c" y "d" de una relación R en la que están presentes las dependencias funcionales "a" → "b" y ("b"+"c") → "d", entonces también estará presente la dependencia funcional ("a"+"c") → "d"

- **Descomposición**: Dados los atributos a, b y c de una relación R en la que está presente la dependencia funcional "a" → "b" y se cumple que "c" ⊆ "b" entonces también estará presente la dependencia funcional "a" → "b"

3.3.2 Reglas de normalización

La teoría de la normalización se basa en las reglas de normalización. Decimos que una relación está en forma normal si satisface este conjunto de restricciones impuestas. La aplicación de una regla de normalización es una operación que toma una relación como argumento de entrada y da como resultado dos o más relaciones:

- La relación objeto de la aplicación de la regla es desestimada en el nuevo esquema relacional
- No se introducen nuevos atributos en el proceso de normalización
- Los atributos de la relación objeto de la normalización pasan a formar parte de la intención de una o más relaciones resultantes.
- En la aplicación de la normalización se deberá eliminar al menos una dependencia existente entre atributos de la relación.

De esta forma, la sucesiva aplicación de las reglas de normalización va a dar lugar a la generación de un número mayor de relaciones y desde el punto de vista lógico una redundancia de los atributos considerados en el esquema.

Por regla general, se dice que un esquema relacional es consistente si las relaciones satisfacen al menos la forma normal de Boyce-Codd.

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

3.3.2.1 Primera forma normal (FN1)

Una relación R satisface la primera forma normal (FNJ) si y solo si, todos los dominios subyacentes de la relación R contiene valores atómicos, es decir, sus atributos solo pueden tener un único valor a la vez.

3.3.2.2 Segunda forma normal (FN2)

Una relación R satisface la segunda forma normal (FN2) si y solo si satisface la primera forma normal y cada atributo de la relación depende funcionalmente de forma completa de la clave primaria de esa relación. Y además no exista una dependencia funcional no completa entre atributos de la relación que forman pare de la clave (Atributos no primos) y clave de la relación.

Si la clave principal es simple, ya está en forma normal, solo nos preocupamos cuando son agregados de atributos, en este caso vemos la dependencia funcional con los demás atributos y no entre los de la clave principal.

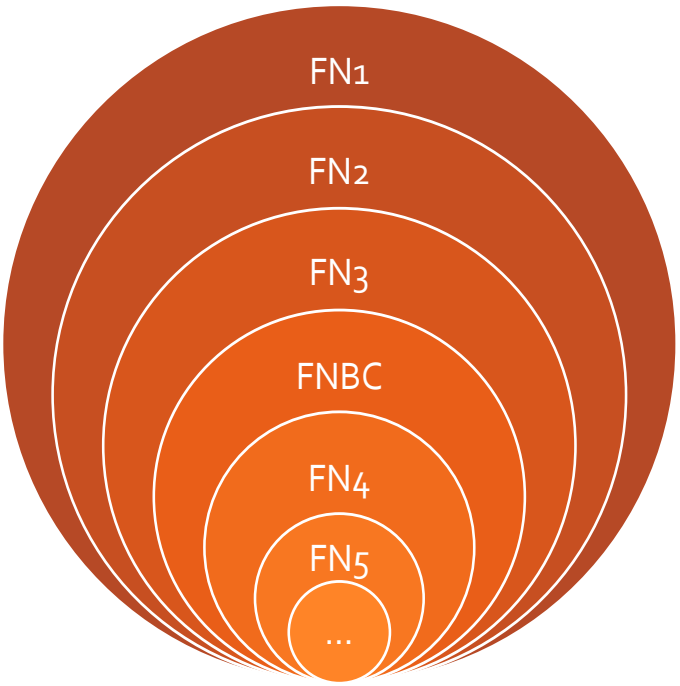
3.3.2.3 Tercera forma normal (FN3)

Una relación R satisface la tercera forma normal (FN3) si, y sólo si, satisface la segunda forma normal y cada atributo no primo de la relación no depende funcionalmente de forma transitiva de la clave primaria de esa relación. Es decir, no pueden existir dependencias entre los atributos que no forman parte de la clave primaria de la relación R.

Un determinante funcional es un atributo simple o compuesto el cual determina funcionalmente de forma completa a cualquier otro atributo de la relación.

3.3.2.4 Forma normal de Boyce-Codd (FNBC)

Una relación R satisface la forma normal de Boyce-Codd (FNBC) si, y sólo si, se encuentra en FN 1, y cada determinante funcional es una clave candidata de la relación R. Se pueden dar casos en los que una relación se encuentre en FN3 pero no satisfaga la FNBC, puesto que la FNBC es más restrictiva que la FN3.



Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

3.3.2.5 Proceso de descomposición (Aplicación de las FN)

El proceso de descomponer una relación no es un proceso trivial y debe ser realizado siguiendo los siguientes principios:

- **Descomposición por aplicación de la FN2**: Dada una relación R cuya clave es un agregado de la forma (**x, y**), y en esta relación existe una dependencia funcional incompleta de la forma "**y**"→ "**z**", donde "**z**" es un atributo no primo de la relación R:
    - Eliminamos de R el atributo z, quedando igual el resto.
    - Construimos R, con intención:
        - "**z**" es atributo no primo de R1
        - "**y**" es clave primaria de R1
    - Definimos "**y**" como clave foránea de la relación R1

  Definimos entonces que esta descomposición debe realizarse en base a la dependencia funcional incompleta y nunca por dependencia entre atributos de la relación.

- **Descomposición por aplicación de la FN3:** Dada una relación R con clave "**x**" y dos atributos no primos "**y**" y "**z**" y en esta relación existen las dependencias funcionales: "**x**" → "**y**", "**y**" → "**z**" por tanto la dependencia transitiva: "**x**" → "**z**" realizamos el siguiente proceso de descomposición:
    - De R eliminamos el atributo que mantiene una dependencia funcional transitiva con la clave de la relación: "**z**" y dejando el resto de la relación igual.
    - Construimos una nueva relación R1 con intención:
        - "**z**" que mantenía una dependencia funcional transitiva.
        - "**y**" con el cual "**z**" mantenía una dependencia funcional completa.
        - La clave de R1 será "**y**" y en esta relación solo existirá una dependencia funcional completa de la forma "**y**" → "**z**"
    - En R se define "**y**" como clave foránea de R1

  Luego la descomposición que se debe realizar por aplicación de la FN3 se debe hacer basándose en la dependencia funcional transitiva y nunca por cualquier otra dependencia entre los atributos de la relación.
- **Descomposición por aplicación de la FNBC:** Dada una relación R en la cual están presentes uno o más determinantes funcionales, "**x**", "y", ...,"**z**" formados por un agregado de daos de la forma: [] ≡ {"**a**", "**b**",..., "**j**"} y en esta relación existen dependencias funcionales distintas a dependencias completas de la forma: []→"**n**" donde "**n**" es cualquier atributo de la relación R, descomponemos:
    - De la relación R se elimina "**n**", quedando igual el resto de la intención.
    - Se construye una nueva relación R1 cuya intención es:
        - "**n**" como atributo no primo de R1.
        - El atributo "**m**" como clave primaria, donde "**m**" $\in$ [], "**m**" $\subset$ [] está formado por un subconjunto de los agregados de datos que forman parte del determinante funcional [] y por el que existe una dependencia funcional de la forma "**m**" → "**n**"
    - Se define "**m**" como clave foránea de la clave de la relación R1.

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

Tema 4: El algebra relacional. SQL

4.1 El álgebra relacional

SQL es un lenguaje de manipulación de datos de los SGBD relacionales basado en el lenguaje algebraico al cual se le ha añadido una semántica que lo hace más próximo al lenguaje natural.

SQL utiliza una serie de operadores algebraicos que operan sobre las relaciones o tablas de un esquema relacional. Cada operador puede operar sobre una o dos tablas (operadores unarios y binarios) pero siempre sobre la totalidad de las tuplas que forman la extensión de la tabla.

El resultado de una operación SQL es una tabla, la cual, a su vez, es susceptible de ser sometida a nuevas operaciones SQL.

4.1.1 Los operadores básicos

Los operadores algebraicos básicos son: unión, diferencia, selección, proyección y producto cartesiano. Los operadores unión, diferencia y producto cartesiano son operadores **binarios**, mientras que los operadores selección y proyección son **unarios**.

La definición para que una tabla sea compatible es la siguiente:

- Dos relaciones R1 y R2 se dice que son compatibles si ambas relaciones tienen el mismo grado y el atributo enésimo de R1 está definido en el mismo dominio que el atributo enésimo de la relación R2, el nombre de estos atributos puede ser distinto.

4.1.1.1 Operador unión (UNION)

La unión de dos relaciones compatibles R1 y R2, es una nueva relación R3, también compatible, cuyo esquema es igual al esquema de R1 y R2, y cuya extensión está formada por la agrupación, sin repetición, de las extensiones R1 y R2.

R1 →

A	B
1	1
1	2
1	3
2	1
2	2
3	1

R2 →

C	D
1	1
1	3
2	1

R3 ≡ R1 + R2 (UNION)

R3 →

A	B
1	1
1	2
1	3
2	1
2	2
3	1

En este caso R3 ≡ R1

SQL→ select * from R1 UNION select * from R2;

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

4.1.1.2 Operador diferencia (MINUS)

La diferencia de dos relaciones compatibles R1 y R2 es una nueva relación R_3, también compatible, cuyo esquema es igual al esquema de R1 y R2, y cuya extensión está formada por aquellas tuplas de la relación R1 que no se encuentran en la relación R2.

$R_3 \equiv R1 - R2$ (MINUS)

$R_3 \rightarrow$

A	B
1	2
2	2
3	1

En algunos sistemas no existe el MINUS, por lo que podemos usar en SQL el NOT EXIST.

SQL→ select * from R1 MINUS select * from R2;

4.1.1.3 Operador selección

La selección sobre una relación R1 mediante una cualificación Q es una nueva relación R2, cuyo esquema es igual a R1, y cuya extensión está formada por todas aquellas tuplas de R1 que satisfacen la cualificación Q.

La cualificación Q es una expresión lógica cuyo formato general es:

R1.atributo ⓧ valor [(AND, OR) R1. ⓧ valor...]

Donde ⓧ representa a los operadores de comparación (≤, <, =, >, ≥, ≠).

Se expresa de la siguiente manera: R_3 = SELECT (R1/Q)

R_3=SELECT (R1/R1.A=2)

$R_3 \rightarrow$

A	B
2	1
2	2

SQL→ select * from R where R.A=2

4.1.1.4 Operador proyección

La proyección sobre una relación R1 con esquema $R1.a_i$, $R1.a_j$, ..., $R1.a_z$ mediante un subesquema $S(R1) \equiv R1.a_m$, $R1.a_n$, ..., $R1.a_p$ donde $a_m \geq a_i$ y $a_p \leq a_v$ es una nueva relación R2, cuyo esquema es igual al subesquema s(R1), y cuya extensión es igual a todas las tuplas de R1 sin repetición sobre el subesquema S(R1).

Se expresa de la siguiente manera : R_3=PROJECT(R1/R1 P)

R_3=PROJECT(R1/R1 A)

SQL→ select R1.A from R1;

$R_3 \rightarrow$

A
1
2
3

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

4.1.1.5 Operador producto cartesiano (PRODUCT)

El producto cartesiano de dos relaciones R1 y R2, no necesariamente compatibles, es una nueva relación R3, cuyo esquema es igual a la concatenación de los esquemas de R1 y R2 y cuya extensión está formada por el conjunto de las tuplas que se obtiene de concatenar cada una de las tuplas de R1 con todas y cada una de las tuplas de R2.

Se expresa de la siguiente manera : R3=R1xR2(PRODUCT) SQL→ select * from R1, R2;

R3 →

A	B	C	D
1	1	1	1
1	1	1	3
1	1	2	1
2	1	1	1
2	1	1	3
2	1	2	1
Etcétera			

4.1.2 Operadores algebraicos avanzados

4.1.2.1 Operador intersección

La intersección de dos relaciones compatibles R1 y R2 es una nueva relación R3, también compatible, cuyo esquema es igual al esquema de R1 y R2, y cuya extensión está formada por el conjunto de tuplas que son comunes a R1 y R2.

Esta operación se forma por un conjunto de operaciones básicas, en particular, por una operación unión y dos operaciones diferencia.

R3= R1 INTERSECT R2 = R2 MINUS ((R1 UNION R2) MINUS R1)

R3=R1 INTERSECT R2

R3 →

A	B
1	1
1	3
2	1

SQL→ select * from R1 intersect select * from R2;

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

4.1.2.2 Operador reunión (JOIN)

La reunión de dos relaciones R1 y R2, no necesariamente compatibles, pero en las que existe al menos un atributo con el dominio común, sobre una cualificación Q, es una nueva relación R3 cuya intención está formada por la concatenación de las intenciones de R1 y R2, y cuya extensión está formada por las tuplas que resultan del producto cartesianos de R1xR2 que satisfacen la cualificación Q.

Dependiendo del tipo de cualificación Q la operación recibe distintos nombres:

- **Equireunión**: En la que la cualificación Q es una expresión que contiene el operador de comparación = (igual).
- **Thetareunión**: En la que la cualificación Q es una expresión que contiene un operador de comparación distinto a la igualdad.
- **Reunión natural**: se representa por el signo | x |, y es una Equireunión sobre todos los atributos comunes existentes entre las relaciones R1 y R2.
- **Autoreunión**: Es una reunión natural sobre una misma relación; es decir, R1 = R2.
- **Semireunión**: Esta operación se realiza entre dos relaciones R1 y R2 sobre una cualificación multiatributo Q en la que puede aparecer cualquier operador de comparación y el resultado es una relación R3 de esquema igual a R1, y cuya extensión está formada por todas aquellas tuplas de R1 que intervienen en la reunión de R1 y R2.

R3= JOIN(R1, R2/ R1.A, R2.C)

R3 →

A	B	C	D
1	1	1	1
1	1	1	3
1	2	1	1
1	2	1	3
1	3	1	1
1	3	1	3
2	1	2	1
2	2	2	1
Etcétera			

SQL(Reunion)→ select * from R1, R2 where R1.A=R2.C;

SQL(Semireunion)→ select R1.A, R2.B from R1, R2 where R1.A=R2.C;

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

4.1.2.3 Operador división

La división de dos relaciones R1 ≡ {x_1, x_2, x_3, …, x_z} y R2 de esquema subesquema de R1 es una relación R3 de esquema igual a la diferencia del esquema de R1 menos el de R2, y extensión igual a todas las tuplas de R1 sin repetición para las cuales está presente toda la extensión de la relación R2.

R1 →

A	B	C
1	1	1
1	1	2
1	1	3
1	2	4
2	1	1
2	1	2
2	1	3
2	2	5

R2 →

B	C
1	1
1	1
1	3
2	4

R3=R1÷R2

R3 →

A
1

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

Tema 5: Traducción de esquemas E-R a esquemas relacionales

5.1 Preparación de los esquemas conceptuales

Como paso previo a la aplicación de las reglas de transformación de esquemas conceptuales a esquemas relaciones (RTCAR), debemos de preparar los esquemas conceptuales con las reglas preparatorias (PRTECAR) que faciliten y garanticen fiabilidad.

Las PRTECAR se basan en la aplicación de la primera formal normal a los objetos que forman parte de los esquemas conceptuales, por lo que eliminaremos los objetos que representen las siguientes anomalías:

- Aquellos atributos correspondientes a los tipos de entidad e interrelación que presenten valores múltiples.
- Aquellos atributos correspondientes a los tipos de entidad e interrelación que sean compuestos

El primer punto consiste en que los atributos existentes en el modelo conceptual solo pueden tomar valores atómicos. Mientras que el segundo hace referencia a la eliminación de agregados de datos o atributos formados por otros atributos más simples.

5.1.1 Eliminación de atributos múltiples

El proceso de eliminación de atributos múltiples es muy simple y consiste en:

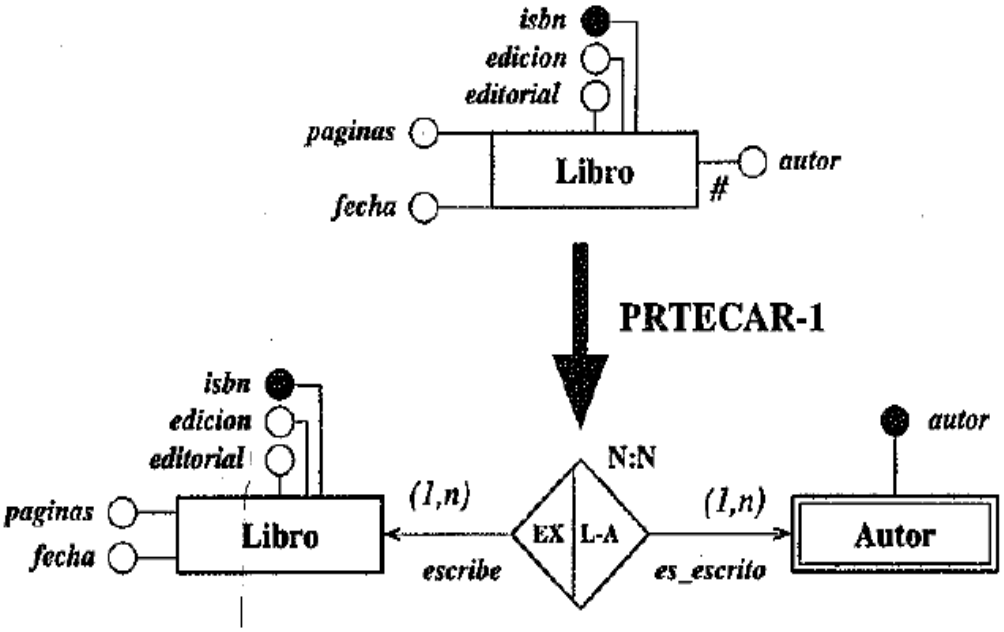
- **PRTECAR-1**: Todos los atributos múltiples; es decir, los atributos que pueden tomar más de un valor en el dominio en el cual están definidos, se transformaran en un tipo de entidad débil por existencia el cual mantendrá una relación:

  - Uno a muchos si el atributo es un identificador alternativo en el tipo de entidad en el que estaba presente.
  - Muchos a muchos en caso contrario

  con el tipo de entidad sobre el cual estaba definido o los tipos de entidad que mantendrán un tipo de interrelación si el atributo múltiple estaba definido sobre el tipo de interrelación.
  Este tipo de entidad débil tendrá como propiedades el atributo al cual se ha aplicado la regla. Además, se deberá tener en cuenta que si el atributo de la entidad débil no identificase sin ambigüedad a las entidades de este tipo entonces procederemos de una de estas formas:

  1. El tipo de entidad débil creada se considera débil por ID con respecto a la entidad con la que mantiene la relación, heredando sus atributos identificadores.

  2. Se añadirá un atributo nuevo que permita identificar sin ambigüedad a las entidades de este tipo de entidad débil.

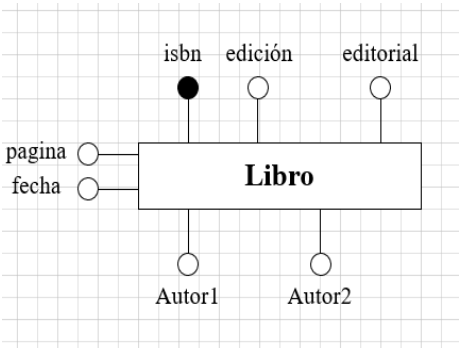


Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

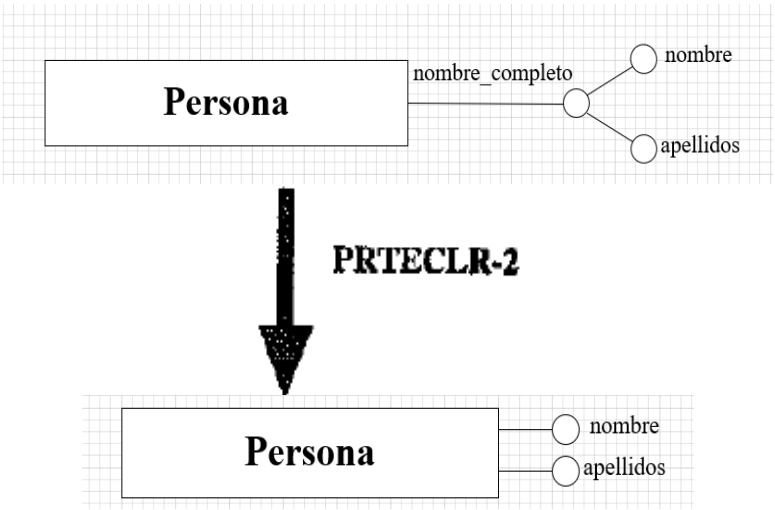
En caso de que sean valores pequeños (Supongamos 2 autores), podríamos representarlos como atributos directos en la entidad libro.

Pero si sobrepasamos cierto valor, se representará como se explicó anteriormente.



5.1.2 Eliminación de atributos compuestos

- **PRTECAR-2**: Todos los atributos compuestos asociados con los tipos de entidad y los tipos de interrelación deben ser descompuestos en los atributos simples que formen parte o intervengan en los atributos compuestos correspondientes. En este proceso de descomposición, se eliminará el atributo compuesto, quedando los atributos simples definidos en el mismo dominio, e interviniendo de la misma forma en el tipo de entidad o interrelación.



5.2 Transformación de los esquemas conceptuales

A continuación, se presentan las reglas RTCAR para la transformación de objetos en los esquemas conceptuales a objetos validos en los esquemas lógicos relacionales. Estas reglas permiten la transformación de los esquemas de uno a otro tipo sin pérdida de infonnaci6n y, por tanto, conservando el nivel de representación del problema.

La aplicación de las reglas va a depender de:

1. El tipo de objeto del esquema conceptual que se debe transformar.
2. La cardinalidad de las relaciones que los objetos mantienen con otros objetos en el esquema conceptual.

5.2.1 Transformación de tipos de entidad

- **RTECAR-1**: Todos los tipos de entidad presentes en el esquema conceptual se transformarán en tablas o relaciones en el esquema relacional manteniendo el número y tipo de atributos, así como la característica de identificador de estos atributos. Por ejemplo:

  Libro (*isbn*. *edición*, *editorial*, *paginas*, *fecha*)

  Autor (*autor*)

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

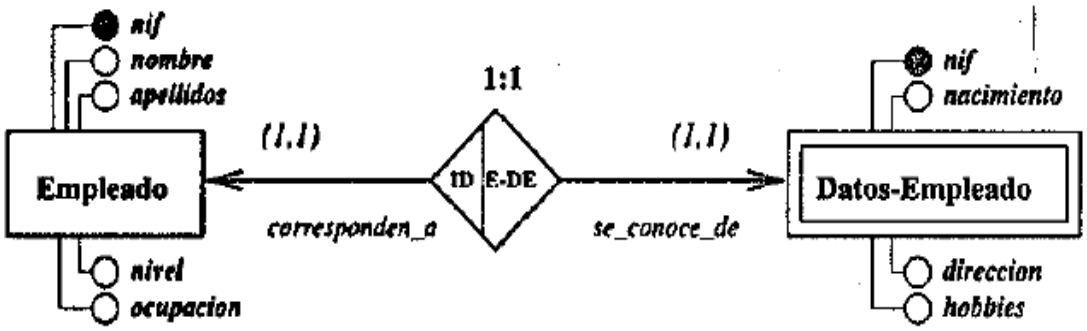
5.2.2 Transformación de tipos de interrelación uno a uno

En base al valor de la cardinalidad mínima de la interrelación identificamos 3 casos:

- (1,1) – (1,1)
- (0,1) – (1,1) / (1,1) – (0,1)
- (0,1) – (0,1)

Por tanto, según cada caso necesitaremos aplicar un tipo de criterio de transformación diferente.

- **RTECAR-2.1**: Si en un tipo de interrelación binaria los dos tipos de entidad participan de forma completa (Ambas tienen cardinalidad mínima y máxima igual a 1), entonces:
    - Si los dos tipos de entidad tienen el mismo identificador:
        - Los tipos de entidad se transforman en una única tabla formada por la agregación de los atributos de ambas.
        - La clave de la tabla es el identificador de los tipos de entidad.
    - Si los tipos de entidad tienen diferente identificador, cada entidad es una nueva tabla y:
        - Cada tabla tendrá como clave principal el identificador de cada uno de los tipos de entidad de los cuales se deriva.
        - Cada tabla tendrá como clave foránea el identificador del otro tipo de entidad con el cual está relacionado.
    - Si los dos tipos de entidad tienen el mismo identificador, pero una de ellas es una entidad débil, entonces se procede de alguna de las dos formas expuestas anteriormente en función de los requisitos funcionales.



Podemos representarlo:

Empleado *(nif, nombre, apellidos, nivel, ocupacion, direccion, nacimiento, hobbies)*

O por cuestiones de procesamiento:

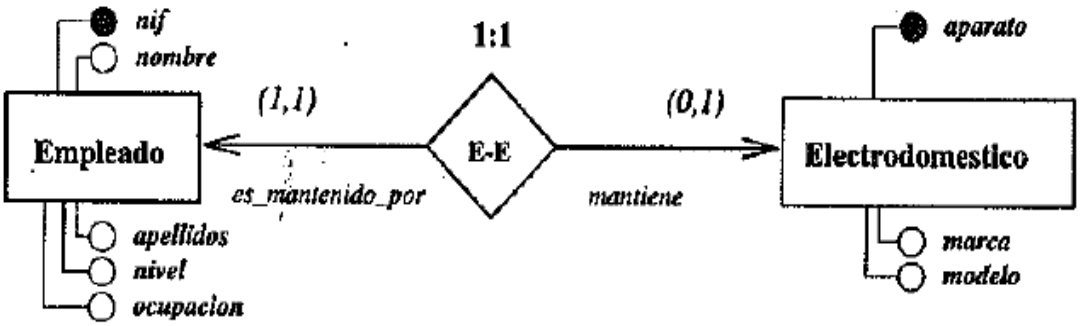
Empleado *(nif, nombre, apellidos, nivel, ocupacion)*
Datos-Empleado *(nif, direccion, nacimiento, hobbies)*

Si existiera un atributo en la relación este podría ir en cualquiera de ambas tablas, pero no en ambas a la vez.

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

Suponiendo el siguiente esquema:



- **RTECAR-2.2** (Participación parcial de una entidad): Si en un tipo de interrelación binaria alguno de los tipos de entidad participa de forma parcial, entonces, cada tipo de entidad se transforma en una tabla por aplicación de la regla RTECAR-1 y se procede de alguna de las dos formas siguientes:
    - El identificador del tipo de entidad que participa de forma total pasa como atributo de la tabla correspondiente a la transformación del otro tipo de entidad. Este atributo será definido como clave alterna y foránea, no pudiendo tomar valores nulos para las diferentes tuplas de esta tabla, y no se genera ninguna tabla para el tipo de interrelación.

      Empleado *(nif, nombre, apellidos, nivel, ocupacion)*
      Electrodomestico *(aparato, marca, modelo, nif)*

    - Se construye una nueva tabla correspondiente al tipo de interrelación formada por los atributos identificadores de los dos tipos de entidad. Los atributos serán claves foráneas de las tablas correspondientes a la transformación de los tipos de entidad.

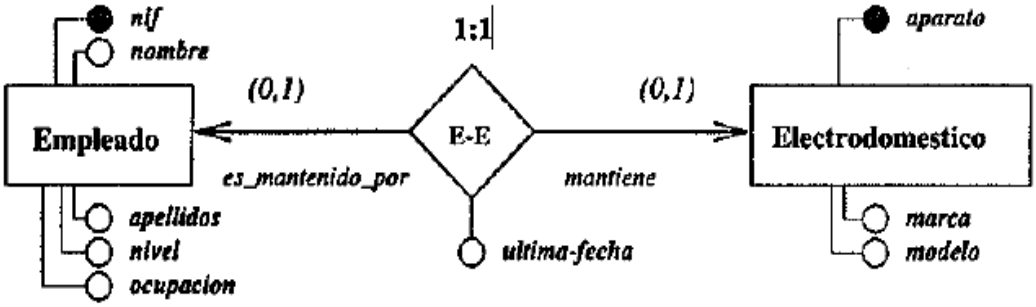
      Empleado *(nif, nombre, apellidos, nivel, ocupacion)*
      Electrodomestico *(aparato, marca, modelo)*
      Emp-Elec *(aparato, nif)*

  La primera trasformación es la más favorable debido a que en ningún momento existirán atributos con valores nulos. Cualquier atributo existente en la relación pasa a la tabla de la entidad con menor cardinalidad mínima.

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

Suponiendo el siguiente esquema:



- **RTECAR-2.3** (Participación parcial en las dos entidades): Por aplicación de la RTECAR-1 cada una de ellas se transforma en una nueva tabla y se procede:
    - Se construye una nueva tabla correspondiente al tipo de interrelación y cuyos atributos serán los identificadores de los dos tipos de entidad, los cuales serán definidos como claves foráneas. La clave principal de la tabla generada será el identificador de uno de los tipos de entidad y se definirá como clave alterna al identificador del otro tipo de entidad.

Empleado	*(nif, nombre, apellidos, nivel, ocupacion)*
Electrodomestico	*(aparato, marca, modelo)*
Empl_Elect	*(nif, aparato, ultima_fecha)*

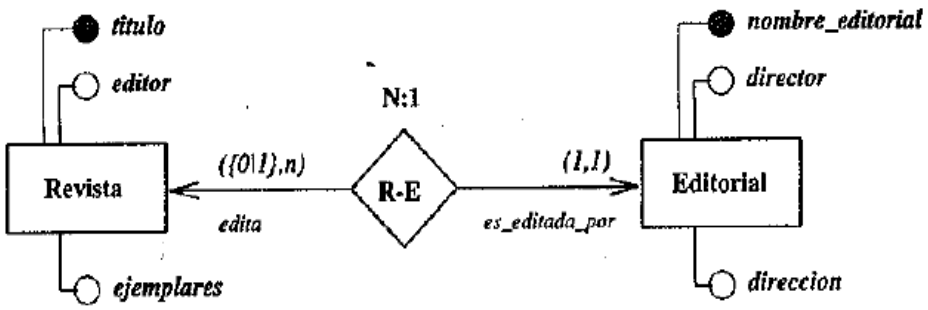
5.2.3 Transformación de tipos de interrelación uno a muchos.

Diferenciamos dos casos:

- (1,1) – (1, n) / (1,1) – (0, n)
- (0,1) – (1, n) / (0,1) – (0, n)

Y según el caso aplicaremos un criterio de transformación:

Suponiendo el siguiente esquema:



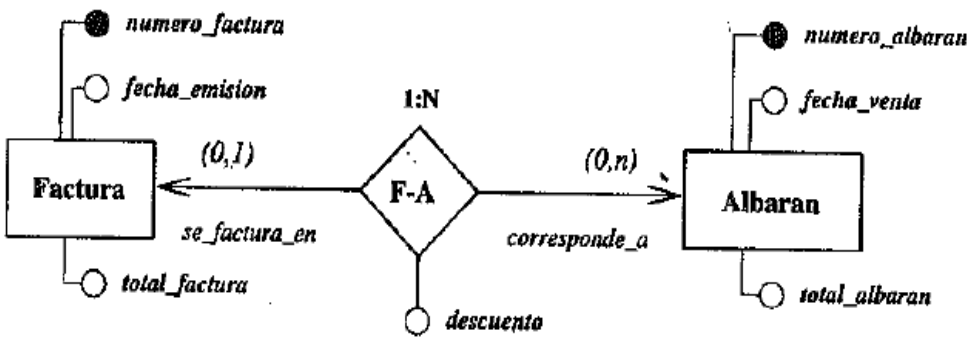
- **RTECAR-3.1**: Si en un tipo de interrelación binaria 1:N ambos tipos de entidad participan de forma total, o el tipo de entidad que interviene con cardinalidad máxima muchos participa de forma parcial, entonces, cada tipo de entidad se transforma en una tabla por aplicación de la regla RTECAR-1, y **el identificador del tipo de entidad que participa con cardinalidad máxima uno pasa a formar parte de la tabla correspondiente al tipo de entidad que participa con cardinalidad máxima muchos**. Este atributo será definido como clave foránea de esta tabla (no pudiendo tomar valores nulos) manteniendo una referencia con la tabla correspondiente al tipo de entidad que participa con cardinalidad máxima uno. Si el tipo de interrelación tuviera atributos asociados, estos atributos pasan a formar parte de la tabla correspondiente al tipo de entidad que participa con cardinalidad máxima muchos.

Editorial	*(nombre_editorial, direccion, director)*
Revista	*(titulo, editor, ejemplares,* **nombre_editorial***)*

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

Suponiendo el siguiente esquema:



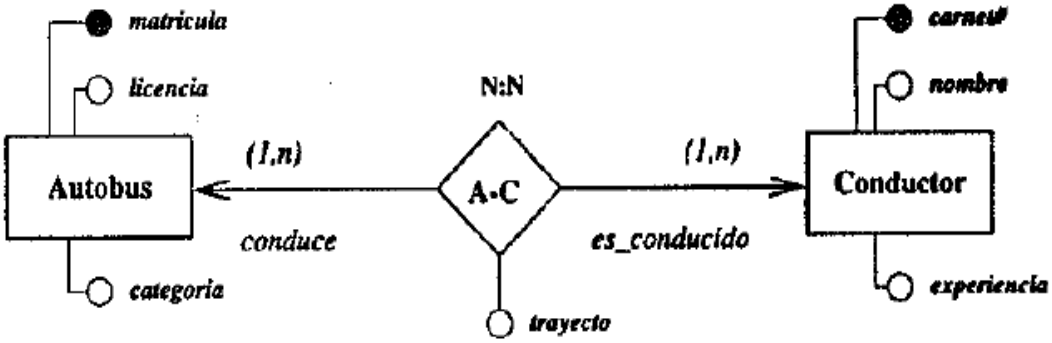
- **RTECAR-3.2**: Si en un tipo de interrelación binaria 1:N ambos tipos de entidad participan de forma parcial, o únicamente el tipo de entidad que interviene con cardinalidad máxima uno participa de forma parcial, entonces, cada tipo de entidad se transforma en una tabla por aplicación de la regla RTECAR-1 y se genera una nueva tabla correspondiente al tipo de interrelación. Esta tabla estará formada por los identificadores de los tipos de entidad que intervienen en el tipo de interrelación y por todos los atributos asociados al tipo de interrelación. La clave principal de esta tabla será el atributo identificador correspondiente al tipo de entidad que interviene con cardinalidad máxima muchos, y será necesario definir como claves foráneas los atributos identificadores correspondientes a los dos tipos de entidad.

Factura	*(numero_factura, fecha_emision, total_factura)*
Albaran	*(numero_albaran, fecha_venta, total_albaran)*
Fact_Alba	**(número_albaran, numero_factura,** *descuento)*

5.2.4 Transformación de tipos de interrelación muchos a muchos

En este caso la cardinalidad mínima es indiferente, se tiene que cumplir que ambas participen con cardinalidad máxima n.

Suponiendo el siguiente esquema:



- **RTECAR-4**: En un tipo de interrelación binaria N:N cada tipo de entidad se transforma en una tabla por aplicación de la regla RTECAR-l y se genera una nueva tabla para representar al tipo de interrelación, estará formada por los identificadores de los tipos de entidad que intervienen en el tipo de interrelación y por lodos los atributos asociados al tipo de interrelación. La clave principal de esta tabla será la agregación de los atributos identificadores correspondientes a los tipos de entidad que intervienen en el tipo de interrelación.

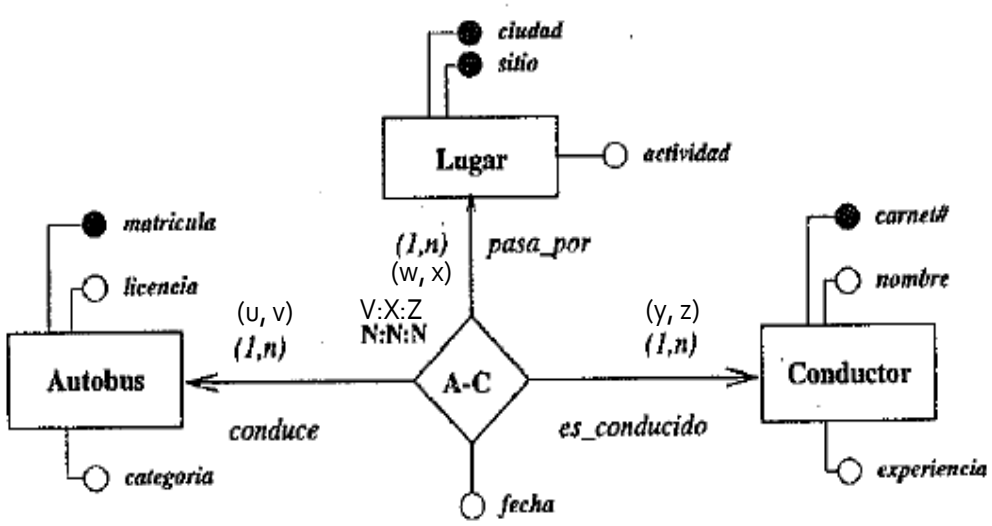
Autobus	*(matricula, licencia, categoria)*
Conductor	*(carnet#, nombre, experiencia)*
Condu_Auto	**(carnet#, matricula,** *trayecto)*

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

5.2.5 Transformación de tipos de interrelación N-arias

Suponiendo el siguiente esquema:



En este tipo de transformación cada tipo de entidad se transforma en una tabla y el tipo de interrelación se transforma en una nueva tabla cuyos atributos son los atributos identificadores de las entidades participantes y los atributos asociados a la interrelación.

Autobus	*(matricula, licencia, categoria)*
Conductor	*(carnet#, nombre, experiencia)*
Lugar	*(ciudad, sitio, actividad)*
Aut_Con_Lug	*(**carnet#, matricula, ciudad, sitio**, fecha)*

Generadas estas tablas debemos de ver la participación de cada entidad, es decir su cardinalidad mínima y su cardinalidad máxima.

- Caso (**V=N, X=N, Z=N**): Este caso es el mostrado anteriormente.
- Caso (**V=N, X=N, Z=1**): Diferenciamos dos casos distintos:
  - La entidad conductor participa con cardinalidad mínima (y) cero. Procedemos de la siguiente manera:

    Autobús **(matricula, licencia, categoría)**

    Conductor **(carnet#, nombre, experiencia)**

    Lugar **(ciudad, sitio, actividad)**

    Aut_Con_Lug **(matricula, ciudad, sitio, carnet#, fecha)**

    Donde carnet# podría ser nulo, dependiendo de los requerimientos del problema los demás atributos deberíamos ver si deberían ser nulos o no.

  - La entidad conductor participa con cardinalidad mínima (y) uno. Procedemos de la siguiente manera:

    Autobús **(matricula, licencia, categoría)**

    Conductor **(carnet#, nombre, experiencia)**

    Lugar **(ciudad, sitio, actividad)**

    Aut_Con_Lug **(matricula, ciudad, sitio, carnet#, fecha)**

    Donde carnet# no podrá ser nula, pero no identificará a esta nueva tabla.

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

- Caso (**V=N, X=1, Z=1**): Dependiendo de los valores de "y" y "w" resolveremos el problema de una manera distinta.

  Donde el modelo relacional sería el siguiente:

  Autobús **(matricula, licencia, categoría)**

  Conductor **(carnet#, nombre, experiencia)**

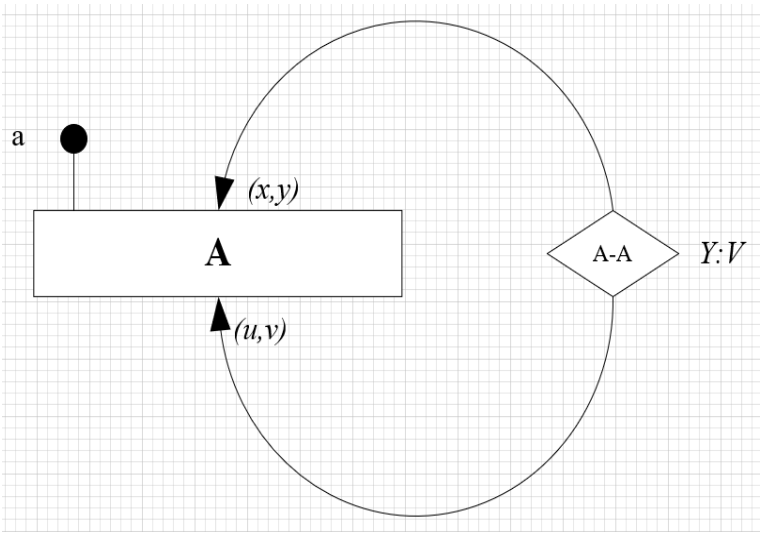
  Lugar **(ciudad, sitio, actividad)**

  Aut_Con_Lug **(matricula, ciudad, sitio, carnet#, fecha)**

  - Caso (**y=0, w=0**): En este caso en la tabla **Aut_Con_Lug** carnet#, ciudad y sitio podrán ser nulos. ← Deducido
  - Caso (**y=1, w=0**): En este caso en la tabla **Aut_Con_Lug** carnet# será siempre no nulo, sin embargo, ciudad, sitio podrá serlo.
  - Caso (**y=0, w=1**): En este caso en la tabla **Aut_Con_Lug** ciudad, sitio será siempre no nulo, sin embargo, carnet# podrá serlo.
  - Caso (**y=1, w=1**): En este caso en la tabla **Aut_Con_Lug** ciudad, sitio y carnet# serán siempre no nulos.

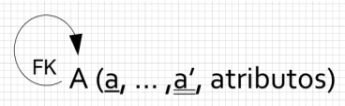
5.2.6 Transformación de tipos de interrelación reflexivas.

Suponiendo el siguiente esquema:



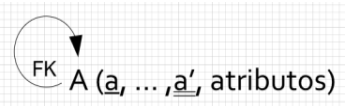
Dependiendo de los valores (x, y, u, v) identificamos distintos caso:

- Caso (**(x, y) = (1,1), (u, v)=(1,1)**): Creamos una tabla por cada entidad existente, la cual tendrá como clave principal "**a**", y como clave foránea la propia clave "**a**" denominada "**a'**" que será clave alterna en la relación. Todos los atributos de la relación irán dentro de esta tabla. Además, que "**a**" sea igual a "**a'**" dependerá del problema.

  Para poder insertar valores en esta relación debemos de tener una transacción que deshabilite las constraints, y que las habilite introducidos los datos.

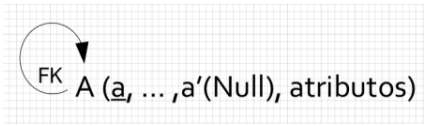
- Caso (**(x, y) = (1,1), (u, v)=(0,1)**): En este caso podemos realizar dos interpretaciones distintas:
  - Si por ejemplo damos el valor de empleados a "**a**", y el valor de jefe a "**a'**" lo representaríamos del siguiente modo:

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

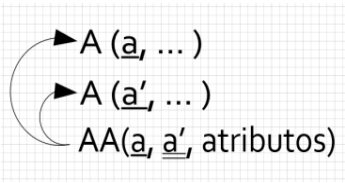
WUOLAH

- En caso de dar los valores contrarios (A "**a**" le damos el valor de jefe y a "**a'**" le damos el valor de empleados) representamos:

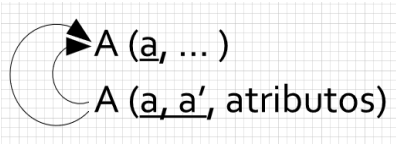


Pudiendo "**a'**" tomar valores nulos (Un jefe puede tener o no tener empleados.)

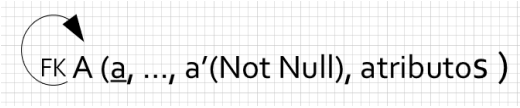
- Caso (**(x, y) = (0,1), (u, v)=(0,1)**): Como participa de forma parcial en la relación realizamos lo siguiente:



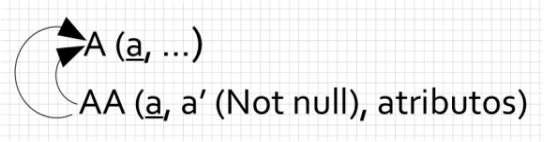
- Caso (**(x, y) = (0,n), (u, v)=(1,n)**): Al igual que en las relaciones N:N se crea una nueva tabla identificada por los atributos de las entidades participantes, y los atributos existentes en la relación.



- Caso (**(x, y) = (1,1), (u, v)=(?,n)**): La ? significa que puede tomar el valor cero como el uno. Para resolver esto planteamos lo siguiente:

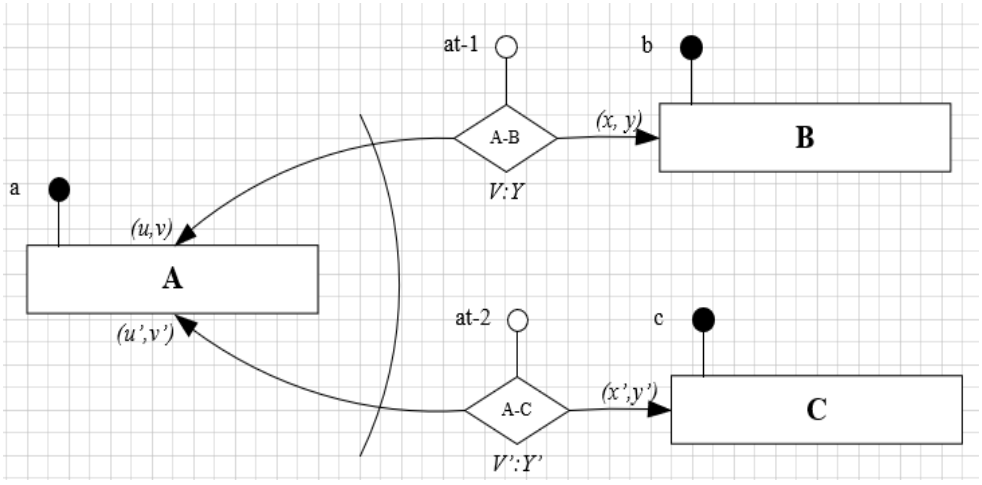


- Caso (**(x, y) = (0,1), (u, v)=(?,n)**): La ? significa que puede tomar el valor cero como el uno. Para resolver esto planteamos lo siguiente:



5.2.7 Transformación de tipo de interrelación exclusiva

Suponiendo el siguiente esquema:



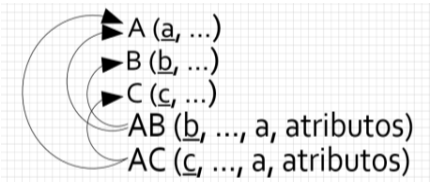
Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

- Caso (**(u, v)=(1,1), (x, y)=(1,n), (u', v')=(1,1), (x', y')=(1,n)**): Cada entidad se convierte en una tabla, y los atributos de "A", van a cada entidad que participe de forma exclusiva.

  A (a, ...)
  B (b, ..., a, at-1)
  C (c, ..., a, at-2)

- Caso (**(u, v)=(0,1), (x, y)=(1,n), (u', v')=(0,1), (x', y')=(1,n)**): Cada entidad se convierte en una tabla, y por cada relación de exclusividad generamos una nueva tabla (por cada relación) donde la clave es la clave de la entidad que participa en la exclusividad:

  Este es bastante complejo, ya que necesitamos de muchas claves foráneas.

- Caso (**(u, v)=(?,n), (x, y)=(1,1), (u', v')=(?,n), (x', y')=(1,1)**):

  A (a, ..., b, c, at-1, at-2)

  Donde los atributos "**b**" y "**c**" podrán ser nulos dependiendo de "**u**" y "**u'**".

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

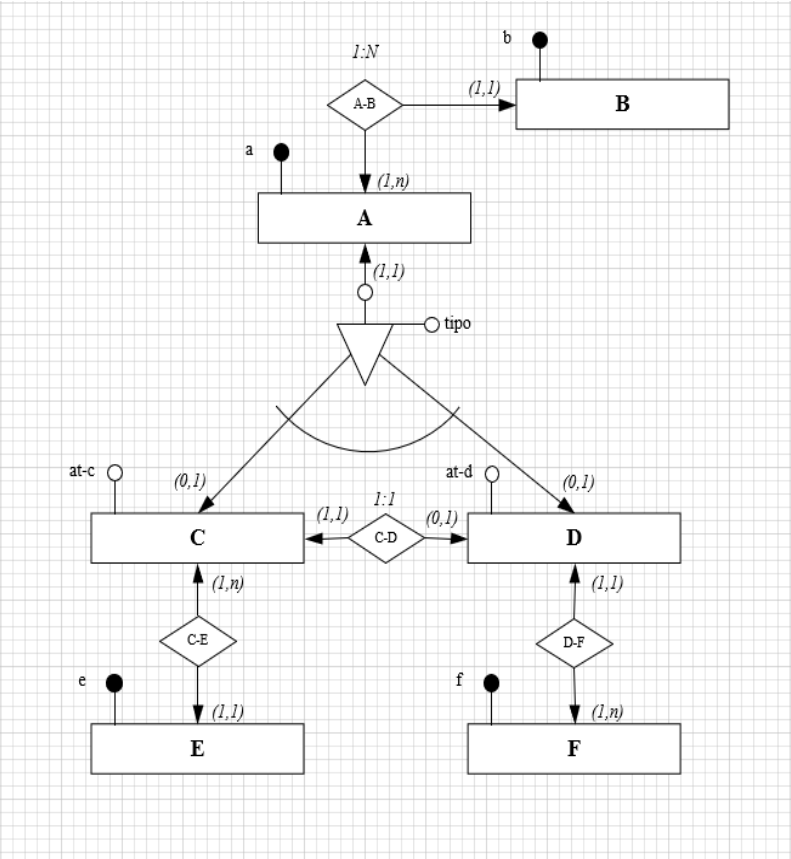
WUOLAH

5.3 Eliminación de las relaciones jerárquicas

Para realizar el proceso de eliminación de las relaciones jerárquicas se deberá aplicar alguna de las siguientes PRTECAR, que elegiremos dependiendo de:

- **Magnitud de la especialización** que los subtipos tienen respecto al supertipo. La especialización supone que los subtipos tienen diferentes atributos asociados que los diferencian.
- El **tipo de especialización que representa** la interrelación jerárquica (Total exclusiva, parcial exclusiva, total inclusiva, parcial inclusiva)
- Otros **tipos de interrelación** que mantengan los subtipos como el supertipo de entidad.
- **Criterios de procesamiento** y, sobre todo, la forma a la que se accederá a la información del supertipo y los subtipos como los tipos de interrelación que mantienen.

Suponiendo el siguiente esquema:



Teniendo esto en cuenta las PRTECAR son las siguientes:

- **PRTECAR-3**: ***Eliminación del supertipo de entidad***. En un tipo de interrelación jerárquica se desestimará el supertipo de entidad, transfiriendo todos los atributos del supertipo a cada uno de los subtipos y cada uno de los tipos de interrelación y cada uno de los tipos de interrelación que mantuviera el supertipo de entidad serán considerados para cada uno de los subtipos, manteniéndose, los tipos de interrelación cada uno de los subtipos de entidad. Además, el atributo calificador del tipo de interrelación, si estuviera presente se puede desestimar.

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

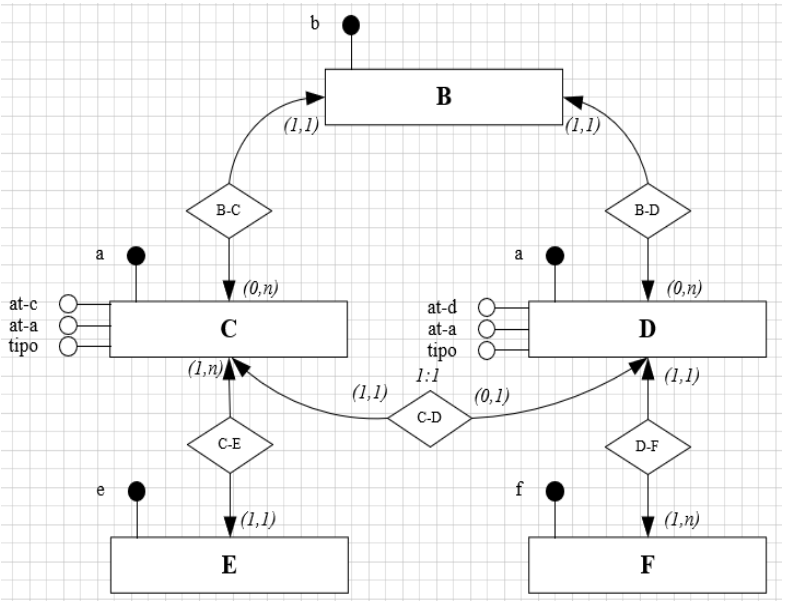
WUOLAH

Por lo que aplicando esta PRTECAR el esquema sería el mostrado a la derecha.

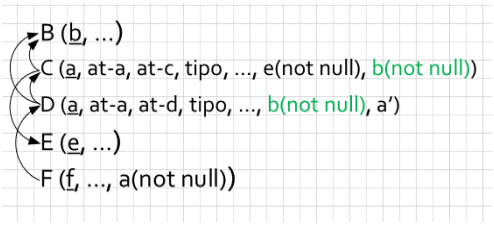
Si el tipo de relación fuera parcial nunca podríamos eliminar el supertipo ya que no existe una tabla que represente todos los posibles valores no especificados, por lo tanto, siempre debe de ser total.

Si la relación fuera inclusiva debemos comprobar que los at-a deben de ser los mismos para los "**a**" de "**c**" y "**d**".

Esta regla es conveniente aplicarla cuando la interrelación jerárquica es total y exclusiva.



El modelo relacional sería el siguiente:

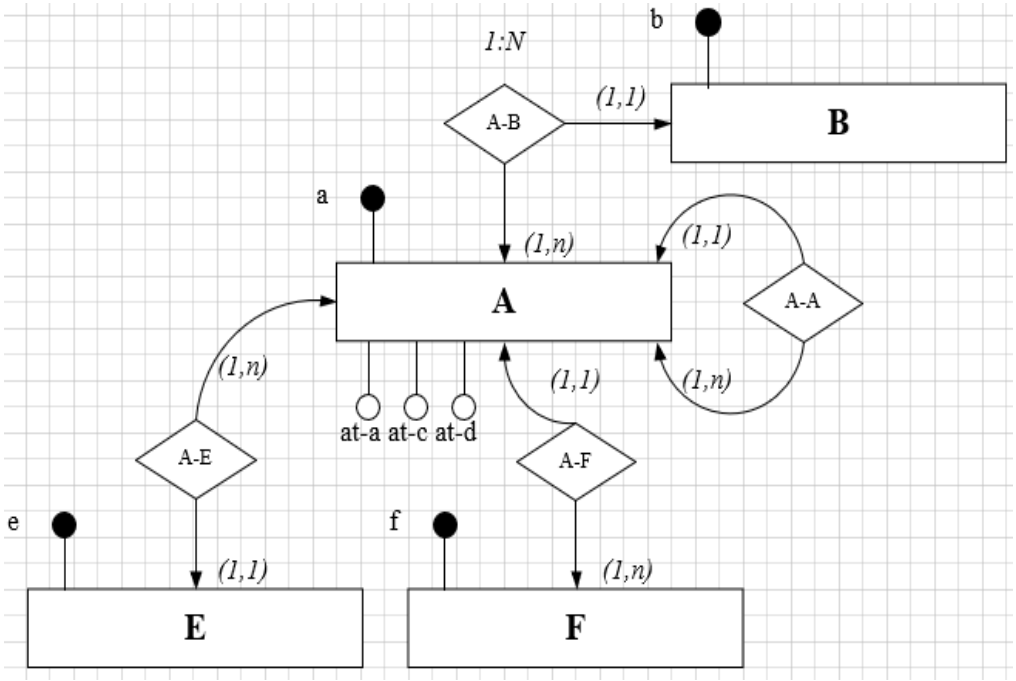


En el cual las partes escritas en verde hacen referencia a cuando la relación es inclusiva.

El atributo tipo, en las tablas c y d, será siempre el tipo de entidad que son (C o D).

- **PRTECAR-4**: ***Eliminación de los subtipos de entidad***. Se eliminan los subtipos de entidad, transfiriéndose todos los atributos de los subtipos al supertipo y cada uno de los tipos de interrelación que mantuvieran los subtipos de entidad serán considerados para el supertipo, manteniéndose los tipos de interrelación en los que intervenga el supertipo de entidad.

  Si el tipo de interrelación jerárquica es exclusivo, el supertipo de entidad participará de forma parcial (cardinalidad mínima cero) en aquellos tipos de interrelación transferidos desde los subtipos de entidad. En caso contrario (inclusiva) participará con las cardinalidades que participaba cada subtipo de entidad en los tipos de interrelación transferidos por aplicación de esta regla.



Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

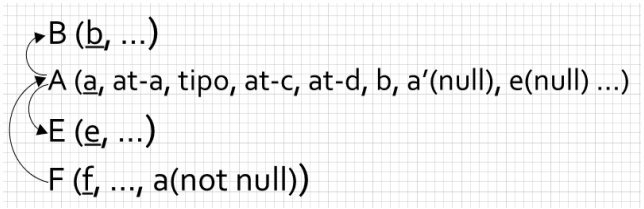
WUOLAH

El atributo cualificador del tipo (atributo "**tipo**") de interrelación pasa a formar parte del supertipo de la siguiente forma:

- Si es exclusivo no formará parte de la clave.
- Si es inclusivo formara parte de la clave formando redundancia de los atributos del supertipo para cada instancia del subtipo.
- Si es parcial, podrá tomar valores nulos para representar a entidades no especializadas.

El uso de esta regla va a dar lugar a un esquema mucho más simple, pero en el caso de tipos de interrelaciones exclusivas y/o parciales se van a presentar muchos posibles valores nulos para aquellos atributos transferidos desde los subtipos al supertipo de entidad.

El modelo relacional sería:

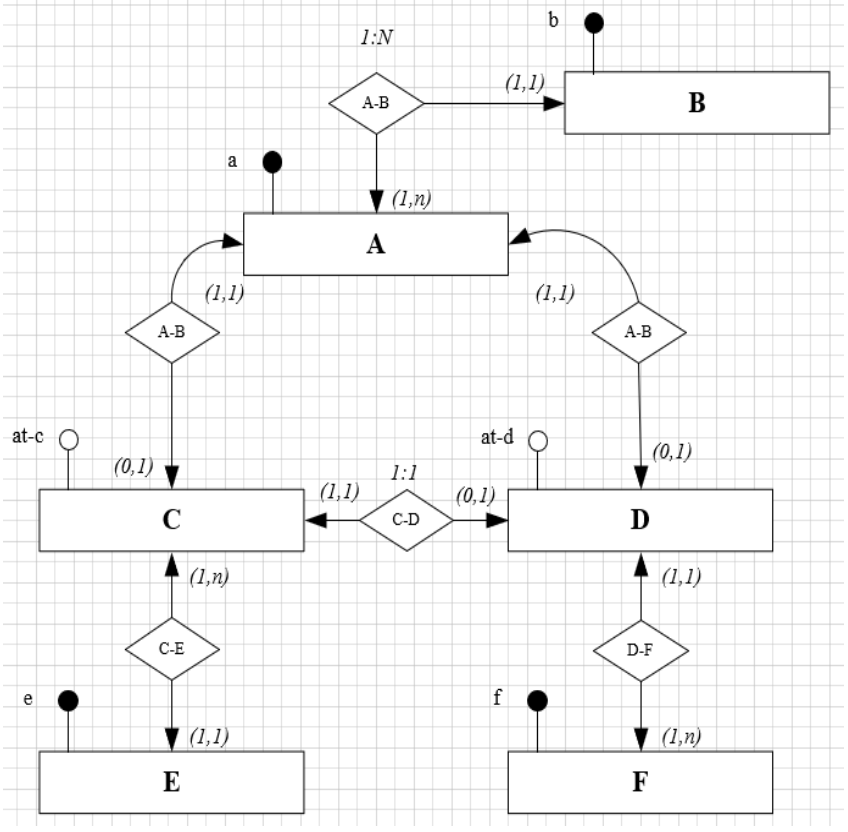


Se puede aplicaren totales o parciales, varía entre exclusivas o inclusivas.

En el caso que el atributo "**tipo**" valga c, el atributo "**at-d**" será null y "**e**" es not null, en caso de que "**tipo**" valga d, entonces "**at-c**" y "**e**" son nulos.

- **PRTECAR-5**: ***Eliminación de la jerarquía***. El tipo de interrelación jerárquica se transformará en tantos tipos de interrelación uno a uno como subtipos de entidad estén presentes, manteniéndose los tipos de interrelación en los que intervienen tanto los subtipos, como el supertipo de entidad. En los tipos de interrelación generados por la transformación, los subtipos participan:
  - Si es exclusiva participan con cardinalidad mínima cero.
  - Si es inclusiva participan con cardinalidad mínima cero o uno.

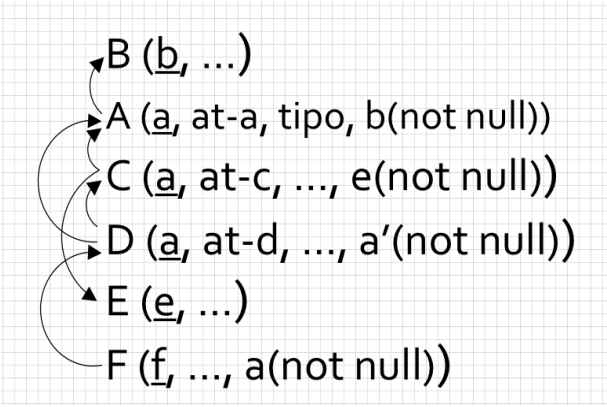
En estos tipos de interrelación el supertipo participa con cardinalidades mínima y máxima igual a uno, pudiendo considerarse que los subtipos de entidad actúan como tipos de **entidad débiles por identificación** con respecto al supertipo.



Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

El modelo relacional sería:



Para insertar un C o D necesitamos previamente de un A con el "**tipo**" correspondiente.

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

Tema 6: Integridad, seguridad y privacidad de las BDD

6.1 Integridad de las BDD

Para mantener la integridad de una BDD el SGBD lo hace mediante lo siguiente:

- Integridad de clave
- Integridad Referencial
- **Integridad de Dominio**: Esta puede ser simplificada mediante la creación de dominios explícitos.
- **Integridad funcional**:
    - **Los asertos**: Regla que se debe de cumplir si o si, puede afectar a una única tabla o a muchas.
      create assertion <nombre-aserto> check <acción>
    - **Los triggers** (Disparadores): Permiten la autogestión de la BD, por ejemplo, si un atributo precio aumenta su valor, que otro campo de la tabla modifique también su valor.
- Existen dos tipos de restricciones:
    - **Deferables**: Se comprueba el resultado después de hacer la acción, para saber si debemos de hacer rollback o commit (Es lo usual).
    - **No deferables**: Se comprueba el resultado antes de realizar la acción, para saber si debemos de ejecutarla o no.
- También podemos comprobar las acciones en el mismo momento que se ejecutan (en lugar de).

Más información respecto a la integridad en el documento de Alberto

Bases de datos activas:

- Modelo ECA: Las reglas que especifican acciones que son activadas de forma automática por determinados eventos han sido consideradas en las bases de datos, principalmente, para la definición de restricciones (constraints).
- El modelo que se ha empleado para especificar las reglas de las bases de datos activas se denomina: Evento-Condición-Acción (ECA)
    - Evento: Generalmente son operaciones de actualización de la base de datos que se aplican explícitamente a la base de datos
    - Condición: Declaración que determina si la regla debe o no ejecutarse. Si se especifica una condición, primero se evalúa ésta, y si su evaluación es verdadera se ejecuta la acción asociada a la regla
    - Acción: Operación que se realiza cuando se evalúa afirmativamente la regla. Suele ser un procedimiento SQL.
- Estas reglas son los triggers. Su finalidad es convertir una base de datos "pasiva" en "activa".

6.2 Recuperación de las Bases de datos

El concepto de recuperación de datos trata de proteger la BD contra fallos lógicos y físicos que puedan destruir los datos de forma total o parcial.

Definimos transacción como una secuencia de operaciones que deben de ejecutarse de forma atómica (O se realizan todas las operaciones o ninguna). Aquellas transacciones que terminan con éxito son grabadas en la BD, las que fracasan realizan una restauración del sistema o rollback.

El componente del sistema encargado de lograr la atomicidad se conoce como administrador de transacciones y las operaciones COMMIT (comprometer) y ROLLBACK (retroceder) son la clave de su funcionamiento.

Las características de una transacción son:

- Atomicidad (Visto previamente)
- Consistencia: Despues de la ejecución de una transacción la BD queda consistente.
- Aislamiento: Una transacción no muestra los cambios producidos hasta que finaliza.
- Persistencia: Una vez finalizada una transacción sus efectos perduran en la BD.

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

- Seriabilidad: El efecto de ejecutar una serie de transacciones de forma concurrente o secuencial debe producir el mismo efecto.

Para conseguir anular y recuperar transacciones, el método más usado consiste en llevar un log, en el cual guardamos toda la información necesaria para deshacer o rehacer las transacciones.

Otra alternativa es manejar dos archivos log, uno con la imagen anterior a las modificaciones y otro con la imagen posterior.

El log es usualmente una pila que una vez llena elimina los registros según van entrando los nuevos.

Un concepto relacionado con los log, es el checkpoint, que permite manejar de forma eficiente el contenido de los archivos log. Realizar checkpoints implica:

- Grabar físicamente el contenido de los buffers de datos en la BD física.
- Grabar físicamente un registro de chechpoint dentro del log.

Los puntos marcados como checkpoint, permiten la recuperación de la base de datos en caliente, es decir, después de la caída del sistema se obtiene la dirección del registro de recuperación más reciente y se recorre el archivo de log desde el punto marcado como checkpoint.

6.3 Concurrencia

En sistemas multiusuario necesitamos de mecanismos para controlar la concurrencia. Se pueden producir inconsistencias por el acceso concurrente.

- **Técnicas de bloqueo**: Basadas en una variable asociada a cada elemento de datos que describe el estado de dicho elemento.
    - Exclusivos: cuando una transacción mantiene un bloqueo de este tipo, ninguna otra transacción puede acceder al objeto hasta que este sea liberado.
    - Compartidos: Cuando una transacción bloquea en este modo, permite que otras transacciones retengan el objeto en bloque compartido. Usada cuando queremos impedir modificaciones mientras se consultan los datos.
    - El algoritmo que se utiliza se llama bloqueo de dos fases.
    - El problema de estas técnicas es que pueden producir interbloqueos, que podemos solucionar:
        - Prevenir el deadlock: Obliga a las transacciones a bloquear todos los elementos por adelantado, en caso de no poder bloquear alguno, se espera hasta volver intentarlo.
        - Detectar el deadlock: Se controla de forma periódica que no se haya producido ninguno. La solución es coger las transacciones víctimas y deshacerlas hasta que desaparezca el deadlock.
    - Una granularidad muy gruesa implica gestionar menor numero de bloqueos, pero retrasa la ejecución de muchas transacciones. Una granularidad muy fina permite mayor concurrencia, pero existe más posibilidad de deadlock.
- **Técnicas de marcas de tiempo**: Las marcas de tiempo son identificadores que se asignan a las transacciones, que se consideran el tiempo de inicio de una transacción. Con esta técnica no existen bloqueos, las transacciones se ordenan en función de su marca de tiempo y se ejecutan o se retrasan.
- **Técnicas optimistas**: Las transacciones acceden libremente a los elementos, y antes de finalizar se determina si ha habido interferencias. Este tipo de técnicas considera que las transacciones tienen 3 fases:
    - Lectura: realizan operaciones sobre copias privadas de los objetos.
    - Validación: Se comprueba si los conjuntos de objetos modificados se solapan entre ellos.
    - Grabación: En el caso de no detectar interferencias se graban las modificaciones.

6.4 Seguridad y privacidad de las Bases de datos

La seguridad de las Bases de Datos abarca:

- Cuestiones éticas y legales al derecho de tener acceso a cierta información.

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

- Cuestiones de política, relacionadas con que información no debe estar disponible al público.
- Cuestiones relacionadas con el sistema, como los niveles del sistema en que deben manejarse diversas funciones de seguridad.

La seguridad de las bases de datos se refiere a la protección frente a accesos malintencionados. Para proteger la base de datos hay que adoptar medidas de seguridad en varios niveles: Sistemas de BD, S.O, red, físico y humano.

En relación al SGBD, debe mantener información de los usuarios, su tipo y los accesos y operaciones permitidas a éstos:

- **DBA**: están permitidas todas las operaciones, conceder privilegios y establecer usuarios.
- **Usuario con derecho a crear, borrar y modificar objetos** y que además puede conceder privilegios a otros usuarios sobre los objetos que ha creado.
- **Usuario con derecho a consultar, o actualizar**, y sin derecho a crear o borrar objetos.
- **Privilegios sobre los objetos**, añadir nuevos campos, indexar, alterar la estructura de los objetos, etc. Los SGBD tienen opciones que permiten manejar la seguridad, tal como GRANT, REVOKE, etc. También tienen un archivo de auditoría en donde se registran las operaciones que realizan los usuarios.
- Otro mecanismo de seguridad que ofrecen los SGBD en **entregar información a los usuarios a través de vistas** (CREATE VIEW)

Las vistas son un medio que proporciona al usuario un modelo personalizado de la BD. Permite ocultar datos que el usuario no tiene necesidad de ver, permitiendo simplificar el uso del sistema y la mejora de la seguridad. Las vistas no necesitan la autorización de recursos. Que tengamos una vista no significa que tengamos todos los privilegios sobre ella. Si creamos una vista que no nos permite conceder ninguna autorización, se deniega la solicitud de creación de vista.

- Los privilegios son de autorización y existen los siguientes:
    - LEER, ESCRIBIR, EJECUTAR, SELECCIONAR, INSERTAR, ACTUALIZAR, REFERENCIAR, INDEXAR
- Papel(ROL): Un papel o rol define a un usuario de la BD, con una serie de autorizaciones sobre la misma. Estos papeles o roles nos permiten simplificar la concesión de privilegios agrupando a los usuarios por roles.

La autenticación es la tarea de verificar la identidad de una persona o software que se conecte a la BD. La forma más simple es el uso de una contraseña.

- Otro método más seguro es el método de clave publica/privada.

Tema 7: Bases de datos distribuidas

Una Base de Datos Distribuida es una base de datos almacenada en varias computadoras que se comunicarán entre ellas a través de diferentes medios de comunicación de forma que funcionan como una sola base de datos. Principalmente se usa el término "sitio" para referirse a cada computadora física que contiene una parte de la bases de datos distribuida.

De este modo, una base de datos distribuida consiste en una colección de sitios en los que cada uno puede participar en la ejecución de peticiones del usuario, accediendo a los datos de un solo sitio o de varios de ellos.

Sitio es el conjunto del hardware y software ligados a una unidad central de proceso capaz de soportar aplicaciones informáticas o bien a un conjunto de estas unidades que trabajen conjuntamente sin estar conectadas por líneas exteriores de comunicaciones.

Una transacción es un proceso que interactúa con el SGBD para llevar a cabo la petición de algún usuario.

Un agente es un proceso que coopera para ejecutar una transacción en un sitio concreto:

- Transacción local: es una transacción que requiere un único agente para poder llevarla a cabo (es decir, sobre el sitio local).
- Transacción global: una transacción que requiere varios agentes para poder satisfacer la transacción solicitada (es decir, sobre varios sitios).

Un SBDD es aquel en el que los sitios trabajarán de forma conjunta para que un usuario de cualquier sitio tenga acceso a los datos de cualquier punto de la red como si de un sistema centralizado se tratara.

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

El Sistema de gestión de bases de datos distribuida (SGBDD) es el software que gestiona el sistema de bases de datos distribuida de igual forma que lo hace el sistema de gestión de bases de datos en los sistemas centralizados.

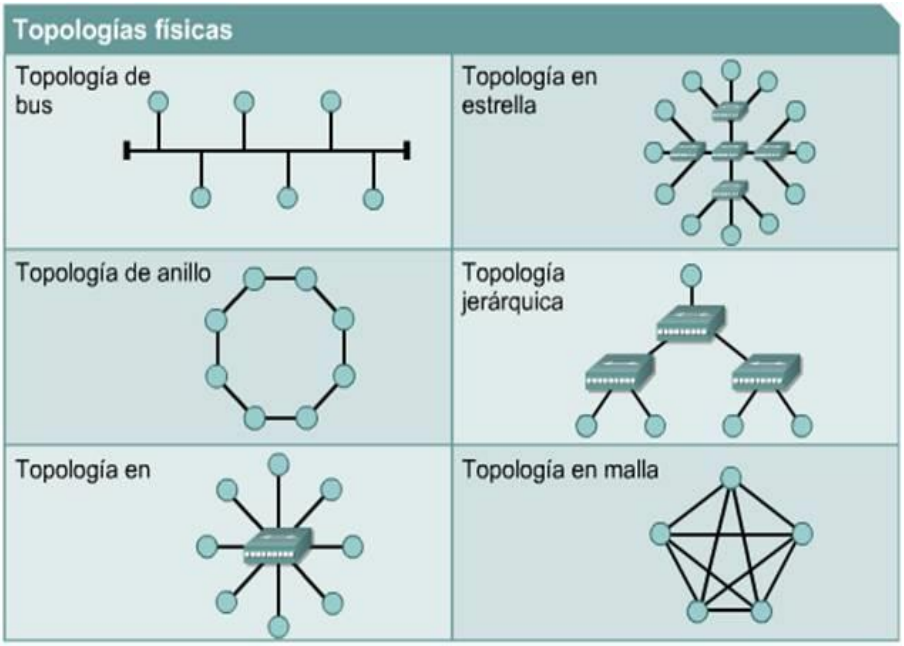
Podemos clasificar las BD según su naturaleza de creación:

- Autonómicas: BD que se diseñan desde un inicio como BD distribuidas con cierta distribución/replicación.
- Federadas: Nacen de la agregación de varias BD centralizadas y que se continúan gestionando de forma relativamente independiente.

Para realizar una distribución de la BD podemos hacerlos en distintas tipologías, que seleccionaremos según lo siguiente:

- Coste de instalación
- Coste de comunicación
- Fiabilidad
- Disponibilidad
- Frecuencia y volumen de datos al que se debe acceder

Las topologías son las siguientes:



Componentes existentes en una base de datos distribuida:

- Los catálogos donde se almacena el esquema global y esquemas externos de los usuarios ligados a la base de datos.
- El coordinador de transacciones que coordina la ejecución de varias transacciones (locales y globales) iniciadas en ese sitio
- El diccionario de datos
- El administrador de transacciones.
- El SGBD local respectivo
- El administrador de la base de datos.

Características:

- Distribución de los datos.
- Flexibilidad de los datos.
- Control distribuido y compartición de datos.
- Fiabilidad y disponibilidad.
- Desempeño de consultas
- Coste, fallos y procesamiento

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

Podemos realizar fragmentación o replicación de datos:

- La fragmentación puede ser de varios tipos:
  - Horizontal-> $R = r_1 \cup r_2 \cup \cdots \cup r_n$
  - Vertical----> $R = r_1 \otimes r_2 \otimes \cdots \otimes r_n$
  - Mixta

Transparencia y autonomía:

- No debe existir dependencia de un sitio central.
- Operaciones continuas (No sé para el sistema para realizar una operación)
- Independencia respecto a la localización, fragmentación, réplica, respecto al equipo, al S.O, a la red y al SGBD.
- Procesamiento distribuido
- Manejo distribuido

El administrador de transacciones tiene como tarea fundamental la transformación de una consulta de alto nivel en otra equivalente que se ejecute con una estrategia más eficiente. Además, deberá calibrar:

- El intercambio de datos entre las diferentes localidades
- La elección del orden de las operaciones relacionales y el mejor sitio para procesar los datos.

El objetivo del procesador de consultas se encuentra el transformar la consulta de alto nivel proporcionada por los usuarios. De entre todas las posibles trasformaciones, escogerá la que minimiza el consumo de recursos.

En un sistema distribuido puede sufrir los mismos tipos de fallos que un sistema centralizado (fallo de memoria, rotura del disco duro, etc.) y además los añadidos con la distribución:

- Fallo de un sitio.
- Fallo de la conexión de red.
- Perdida de mensajes.
- División de la red (descomposición de la red en partes por la pérdida de algún enlace intermedio).

Para la recuperación de fallos:

- Si hay réplicas en el sitio recuperado, hay que actualizarlos antes de que ninguna consulta pueda acceder a los datos sin actualizar.
- Si hay transacciones ejecutándose en un sitio y éste falla, estas transacciones tienen que abortarse tan pronto como se pueda.
- Si el sitio que ha caído es un servidor central para algún subsistema, debe lanzarse una elección para determinar el nuevo servidor

El catálogo es el elemento del sistema que contiene la información de la base de datos y los datos necesarios para el control distribuido de los mismos en los diferentes sitios. Este elemento puede almacenarse de diferentes formas:

- Centralizada. (Depende de un sitio central)
- Replicas completas. (Falta de autonomía)
- Dividido: Cada sitio almacena el catálogo de su contenido local. Así el catálogo global se obtendría de la unión de todos los catálogos. Las operaciones no locales serían muy costosas por las búsquedas necesarias en los sitios.
- Combinación de centralizado y dividido: Cada sitio mantiene su propio catálogo local y además un sitio central único mantiene una copia global centralizada. Mejora el anterior, pero viola la no dependencia de un sitio central, ya que se accede al unificado y de él se obtiene la dirección.

La propiedad de la atomicidad de una transacción en los SGBDD es mucho más complicada, ya que varios sitios pueden estar participando en la ejecución de esa transacción.

Las actividades de la transacción pueden tener lugar en diferentes sitios y puede ser difícil de mantener un orden de tiempo entre las acciones. Por lo que el control de la concurrencia generalmente está basada en el bloqueo.

El coordinador puede ser:

- Coordinador único:

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

    - Fácil implementación
    - Manejo sencillo del interbloqueo, en el que puede usarse prevención de interbloqueo o detección de interbloqueo.
    - Cuello de botella en las peticiones de actualización.
    - Vulnerabilidad: si el coordinador cae, no puede seguirse y hay que elegir otro.
- Coordinador múltiple

El tipo de protocolo puede ser:

- Protocolo mayoritario
  – Mayor dificultad de Implementación: Requiere muchos más mensajes para el bloqueo y para el desbloqueo.
  – Manejo de interbloqueo: tiene el fallo de que puede darse el interbloqueo en situaciones en las que el número de transacciones sea par.

- Protocolo parcial (favorece bloqueos compartidos)
  – Bloqueos compartidos: se pide el bloqueo del ítem al manejador de bloqueo del sitio que contiene la réplica del ítem que deseamos.
  – Bloqueos exclusivos: solicitan el bloqueo a todos los sitios que tienen réplica del ítem requerido.

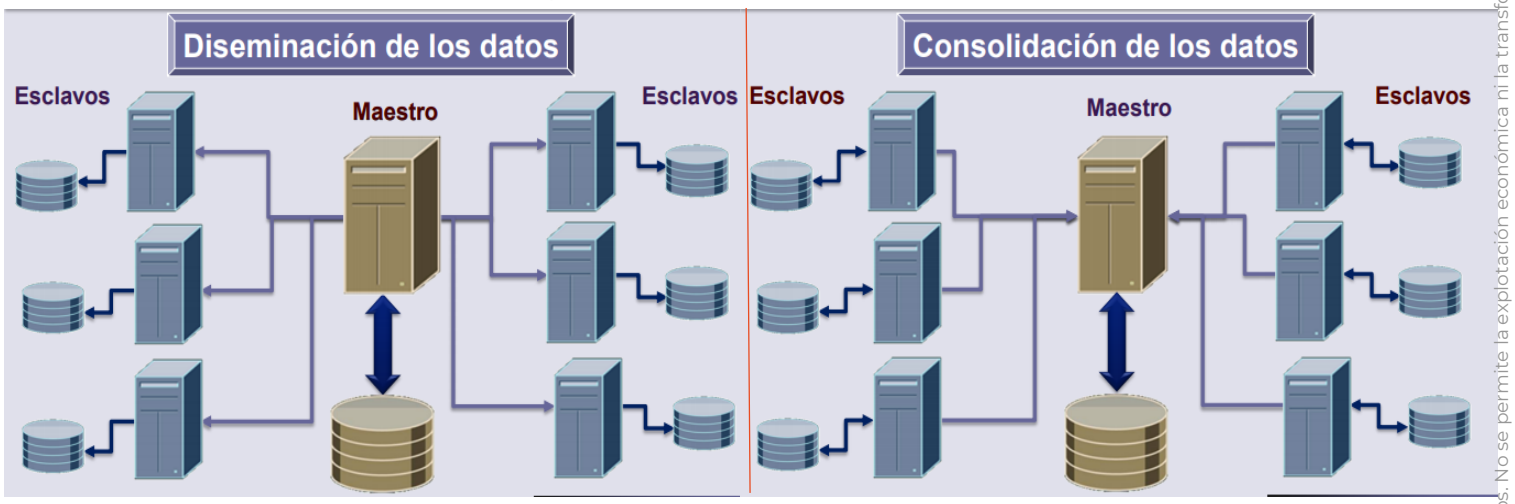
Tema 8: Bases de datos replicadas

- Proceso consistente en la **copia y mantenimiento de objetos de la base de datos en múltiples bases de datos que forman un sistema de bases de datos que utiliza tecnología de bases de datos distribuidas** para compartir datos entre múltiples sitios. Permite un acceso a los datos en todo lugar y en todo momento.
- Los cambios aplicados a un nodo se capturan y almacenan localmente antes de reenviarlos y aplicarlos a cada una de las ubicaciones remotas.
- La principal **diferencia entre una Base de datos replicada y una distribuida** es que en la distribuida los datos están disponibles en muchas ubicaciones, pero cada relación concreta solo se encuentra en una ubicación, a diferencia con las replicadas donde están disponibles en múltiples ubicaciones.
- Los **beneficios** que encontramos en la replicación:
    - **Disponibilidad**: Los datos están disponibles en distintas ubicaciones, lo que permite un acceso a usuarios y aplicaciones mediante opciones de acceso.
    - **Fiabilidad**: Al haber múltiples copias de los datos, se dispone de mecanismos excelentes de recuperación de los datos (en caliente) en caso de fallo.
    - **Rendimiento**: Se incrementa el desempeño de las operaciones de consulta
    - **Reducción de la carga**: El acceso se distribuye entre diferentes servidores. El coste disminuye, pues se puede acceder al servidor cuyo coste de acceso sea menor.
    - Procesamiento **Offline**: La replicación se puede implementar haciendo uso de instantáneas, que son copias completas o parciales de la base de datos en un momento dado (snapshot). Se trabajan sobre ellas de forma offline y cuando se conecta se sincroniza.
    - Soporte **multiusuario**: La replicación permite generar muchas instantáneas personalizadas para las necesidades específicas de cada usuario.
    - Soporta aplicaciones avanzadas:
        - OLTP (OnLine Transaction Processing)
        - OLAP (Online Analytical Processing)
        - Data Mining
        - Mobile Computing

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

- Algunos de los **inconvenientes** son:
    - Mayor complejidad del SGBD que los sistemas centralizados
    - Sobrecarga debido a las operaciones de actualización
    - Mayor complejidad para la gestión del control de concurrencias y recuperación de la base de datos
- Las **aplicaciones** de la replicación son las siguientes:
    - Sincronización limitada (Equipos de ventas remotos).
    - Sincronización continua (Aplicaciones financieras. Gestiones bursátiles)
- **Componentes básicos**:
    - Objetos de replicación: Objeto de una base de datos que existe en múltiples sitios.
    - Grupos de replicación: Conjunto de objetos de replicación lógicamente relacionados, facilitando la administración de los objetos y pueden existir en múltiples sitios de replicación.
    - Sitios de replicación (Respecto a los grupos):
        - Maestros: Controlan uno o más grupos, manteniendo y propagando copias del grupo de replicación a los sitios esclavos. Se comunican entre ellos para propagar los cambios realizados.
        - Esclavos: Pueden contener copia del grupo o parte de ella, contienen al menos una instantánea del grupo de replicación.
- **Entornos de la replicación**:
    - **Síncrona**: Los datos replicados se actualizan de forma inmediata al actualizar los datos originales. Hace uso del protocolo de confirmación en dos fases. Supone una carga elevada en la red y los nodos siempre han de estar online.
    - **Asíncrona**: Hay un retardo entre la actualización de datos origen y destino (minutos, horas, días). Organizaciones que trabajan con replicas que no necesitan estar actualizadas.
- **Propiedad de los datos**: Información acerca del sitio que mantiene el privilegio de actualización de los datos
- **Propiedad maestro/esclavo**: Los datos son asíncronamente replicados por el sitio maestro.
    - Publicación/suscripción
        - El sitio maestro publica los datos y los sitios esclavos se suscriben, recibiendo copias de sólo lectura.
        - Potencialmente cada sitio puede ser maestro de una serie de conjuntos de datos no solapados.
        - No existen copias de datos iguales en diferentes sitios maestros.
            - Un maestro contiene la extensión completa de una tabla o La extensión de una tabla es fragmentada entre distintos maestros.
                - Replicación asimétrica

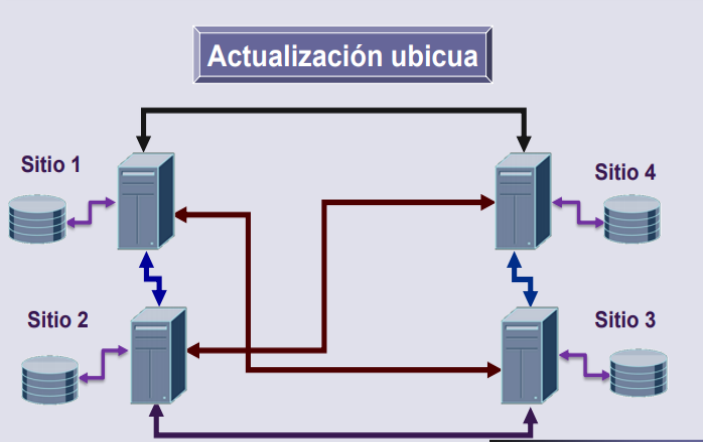


- **Propiedad Flujo de trabajo**: Al igual que maestro/esclavo impide los conflictos de actualización, aunque es más dinámico
    - Permite que el privilegio de actualización se transfiera entre sitios, aunque en cada momento sólo un sitio tiene ese privilegio. Ejemplo: el procesamiento de pedidos, donde el procesamiento de los pedidos sigue una serie de pasos: introducción del pedido, aprobación del crédito, facturación, etc.

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH

- **Propiedad de tipo de actualización ubicua (replicación simétrica)**: Entorno igualitario donde los múltiples nodos disponen de los mismos derechos para actualizar los datos replicados
    - Permite que los sitios locales operen de forma autónoma, incluso cuando otros sitios no están disponibles
    - Puede conducir a conflictos
        - Incorporan procedimientos de detección y resolución



- Servidor de replicación: Es una técnica alternativa y potencialmente más eficiente que la distribución de los datos. Permite la actualización síncrona y asíncrona.
    - **Escalabilidad**: debe gestionar la replicación tanto de pequeños como de grandes volúmenes de datos.
    - **Mapeado y transformación**: debe ser capaz de gestionar la replicación entre distintos SGBD y plataformas
    - **Replicación de objetos**: debe ser posible replicar objetos además de los datos (índices, disparadores, etc.)
    - **Especificación del esquema de replicación**: debe proporcionar mecanismos para permitir que un usuario con privilegios especifique los objetos y datos que se desea replicar.
    - **Mecanismo de suscripción**: debe proporcionar un mecanismo que permita la inicialización de una réplica de destino.
    - **Fácil administración**: simplicidad para el DBA de administrar el sistema, comprobar el estado, y monitorizar el rendimiento de los componentes del sistema de replicación.
    - **Problemas**:
        - Actualizaciones transaccionales: La estructura de la transacción en la base de datos origen se mantiene en la base de datos destino.
        - Instantáneas: Permiten la replicación asíncrona del estado de la base de datos origen de acuerdo a una planificación predefinida (calendario)
            - Utilizan el registro de operación de la base de datos para detectar las modificaciones en los datos origen
            - Implementan una gestión de colas para las actualizaciones de las replicas
        - Disparadores: Es responsabilidad del usuario generar código dentro de un disparador que se ejecute cada vez que se produzca un suceso determinado
            - Técnica más flexible que las instantáneas pero que presenta problemas:
                - Consumo de recursos adicionales
                - No pueden planificarse. Se ejecutan siempre que se produce el suceso
                - Complejo cuando se actualizan muchas réplicas
    - Detección y resolución de conflictos
        - Envío de datos antiguos y datos nuevos
            - Comprobación en el sitio esclavo
        - Se han propuesto diferentes mecanismos:
            - Marcas temporales inferiores y superiores
            - Prioridad de los sitios o Actualizaciones aditivas y promediadas (los datos satisfacen esta propiedad)
            - Valores mínimo y máximo
            - Definido por el usuario
            - Resolución manual
        - Se han propuesto diferentes soluciones
            - Nominación sitio + valor duplicado
            - Codificar (número de secuencia) el valor duplicado
            - Desestimar el valor duplicado

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

WUOLAH