

## Unidad Didáctica VI.- Diseño de BD relacionales

En **Si Andalucía** tienen entre manos el desarrollo de una aplicación que gestione la base de datos sobre empleados, proyectos y departamentos implicados de una empresa. Tienen especial empeño en hacerla bien, porque como su propia empresa va creciendo, cada vez trabajan en más proyectos simultáneamente, y tienen más personas implicadas en su desarrollo. Por eso les interesa conseguir que el software sea altamente reutilizable, de forma que a partir de la aplicación desarrollada para el cliente, puedan desarrollar la suya propia con un esfuerzo mínimo de adaptación. Hacer bien el modelado de la BD es fundamental.



A **Víctor** le han dado las restricciones semánticas del sistema a modelar, y es el encargado de diseñar y poner en funcionamiento la base de datos, con la supervisión de **María** y **Carmen**, como siempre.

Tiene claro lo que tiene que hacer:

Partiendo de la descripción del sistema, hacer un análisis para obtener el diagrama para el modelo Entidad Relación,

obtener a partir de él el esquema relacional,

asegurarse de que las tablas que obtenga en este esquema estén normalizadas para que el diseño sea óptimo,

y finalmente traducir en sentencias SQL todo ese diseño, para implementar de forma efectiva la Base de Datos sobre un SGBD concreto.

Ha realizado el proceso completo, incluso sin demasiada ayuda, en varios proyectos de poca envergadura, pero este nuevo proyecto se le antoja más complicado, y se desanima pensando la cantidad de trabajo que le queda por delante. Le comenta a **María** lo difícil y pesado que es hacer "a mano" todo ese proceso, y **María** y **Carmen** se ríen. **Víctor** se da cuenta de que se ha perdido algo, y les pregunta qué pasa.



**María:** Creo que ya sabes suficientemente bien lo que hay que hacer, como para que dejemos que la informática lo haga por ti. Carmen se va a encargar de explicártelo.

**Carmen:** Existen multitud de Herramientas CASE que facilitan enormemente ese trabajo. De hecho, su uso es lo que permite que aumentemos la productividad, y la calidad del software que producimos, obteniendo gran parte del código y de la documentación de forma automática, además, eso permite reducir nuestros costes, y nos hace competitivos.

**Víctor:** ¿Quieres decir que hasta ahora todo el trabajo que yo he hecho en otras aplicaciones ha sido inútil, que todo lo hace automáticamente un programa?



**Carmen:** No del todo. Estas herramientas generan el código necesario, pero a partir de unas especificaciones de diseño que tú les tienes que suministrar, y para hacer correctamente las especificaciones, debes conocer bien todo el funcionamiento del proceso. No es inútil lo que has aprendido hasta ahora. Sólo que a partir de ahora, podrás centrarte en las tareas imprescindibles, y dejar las rutinarias para que las hagan las herramientas CASE.

**Víctor:** ¿Y usamos alguna de esas herramientas en esta empresa?

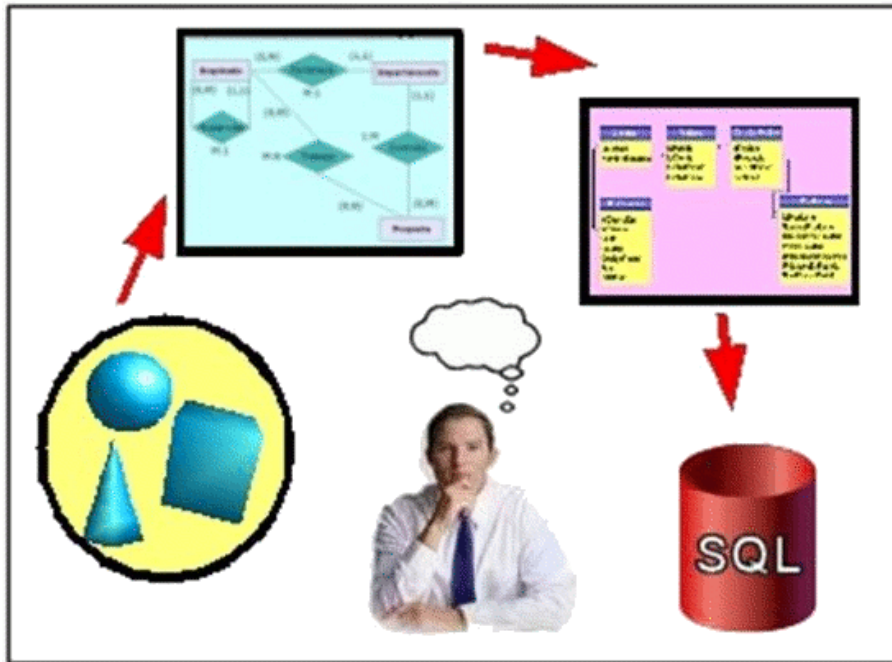
**Carmen:** Varias, de hecho, y son muchísimas las disponibles en el mercado, pero tú vas a comenzar con una llamada TOAD Data Modeler, que es bastante intuitiva, y te va a facilitar el aprendizaje. ¿Listo para seguir aprendiendo?

**Víctor:** Siempre deseoso de aprender, y si es para facilitarme el trabajo, con más motivos.

técnicas utilizadas para, tras analizar una situación del mundo real, llegar a obtener una **representación modelizada** de la realidad que permita su tratamiento informático por un sistema gestor de bases datos.

Recordarás que empezamos aplicando el **modelo Entidad-Relación** para obtener un **modelo lógico de datos**, que posteriormente se traducía a un **esquema de tablas** siguiendo el **modelo relacional**, que a su vez eran **normalizadas**, y que por último **se implementaban en el sistema gestor de bases de datos por medio de un lenguaje estándar como es SQL**.

Esquemáticamente lo ilustrábamos por medio de la siguiente figura, donde tras el análisis de una situación dada, ésta iba tomando forma sucesivamente en un **modelo E/R**, un **esquema relacional basado en tablas normalizadas** y un **conjunto de sentencias SQL** que interactuaban con el sistema gestor de bases de datos.



Como ya sabes, **el modelo E/R, el modelo relacional y el modelo SQL, no son más que distintas maneras de representar la misma realidad**, empezando por el nivel más elevado, conceptualmente hablando, que es el modelo E/R y descendiendo hasta el nivel más ligado al sistema informático que es el SQL.

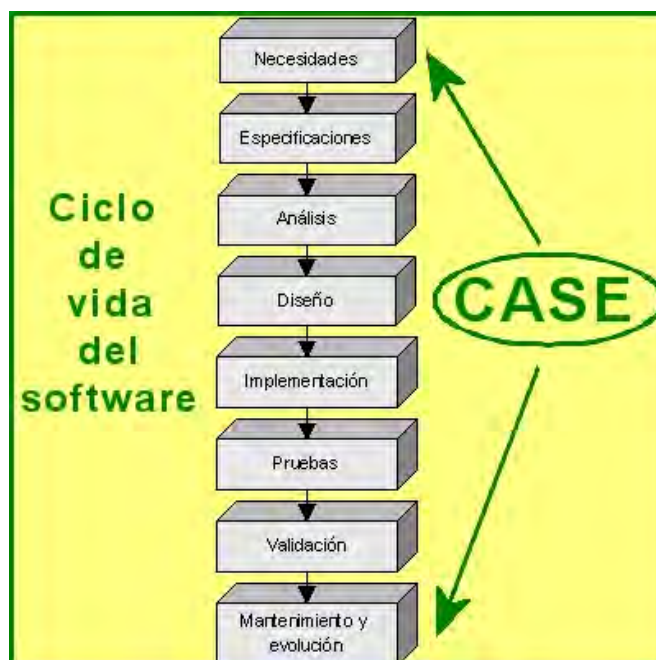
Hasta ahora las únicas herramientas que nos ayudaban en la tarea de representar gráficamente y tratar los distintos modelos eran el lápiz y el papel (el teclado en el caso de SQL). Esto es suficiente para pequeños modelos con pocas entidades y relaciones (pocas tablas en términos relacionales), pero se vuelve excesivamente laborioso, complicado de realizar y mantener, y muy poco práctico y operativo en situaciones con decenas o quizás cientos de entidades y relaciones entre entidades.

Te preguntarás... **¿Hay alguna forma de automatizar la representación de los modelos de datos para aumentar nuestra productividad y permitir centrarnos en las tareas de análisis, dejando la parte de la representación del modelo al ordenador?** Como puedes imaginar la respuesta es sí, la informática viene en ayuda de los propios informáticos para ayudarles en su trabajo, las siglas esta vez son **CASE (Computer Aided Software Engineering)**, **ingeniería software asistida por ordenador**.



**Un sistema CASE es una herramienta informática que permite a los desarrolladores de software concentrarse en las tareas que les son propias, aumentando su productividad y mejorando la calidad de su trabajo. El objetivo máximo que persiguen es la generación automática de programas de ordenador a partir de las especificaciones de diseño.**

En la práctica las aplicaciones CASE cumplen este objetivo en mayor o menor medida. Existe una gran variedad de sistemas CASE que abarcan diferentes fases del ciclo de diseño, desarrollo y pruebas de software ([ciclo de vida del software](#)).



En esta unidad **nos centraremos en aquellas herramientas CASE que ayudan en el diseño y modelado de bases de datos**, de forma que podremos aplicar todo lo estudiado hasta ahora en las unidades anteriores de este módulo formativo. En unidades posteriores conocerás otras utilidades CASE, por ejemplo para generar aplicaciones informáticas completas.

### Autoevaluación

Señala las respuestas correctas. CASE es...:

- a) Computer Aided Software Engineering
- b) Diseño de Programas por Ordenador
- c) Ingeniería Software Asistida por ordenador
- d) Computer Asisted Software Especification

Comprobar

#### Para saber más

*El tema de la ingeniería del software y las herramientas CASE es muy amplio y complejo, constituyendo una parte importante de la informática profesional de hoy día. Te facilitamos dos enlaces que pueden ayudarte a situar mucho mejor estas disciplinas en su contexto e importancia actuales. Pertenecen a la Escuela Superior Politécnica de Albacete.*

[Asignatura de Ingeniería del Software](#)

[Introducción a la ingeniería del software](#) [Versión en caché]

[Herramientas CASE](#) [Versión en caché]

Diseño de bases de datos relacionales

#### CASO.

*Estimulado por el aprendizaje de una nueva herramienta, Víctor, ya está deseoso de trastearla y hacerse con su funcionamiento. Ha estado preguntándole cosas a Carmen, que ha procurado*



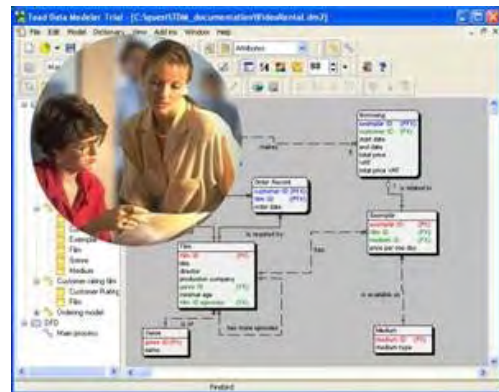
contestarle de la forma más sintética posible, de forma que él tenga las nociones elementales sobre las características que vienen a incluir casi todas las herramientas CASE usadas en el diseño de BD, lo que nos permiten hacer, lo que nos ahorran hacer, y lo que nosotros les tenemos que proporcionar. Algo en lo que Carmen le ha insistido mucho es en que estas herramientas facilitan las labores de análisis, diseño, documentación, e incluso codificación, pero que en ningún caso sustituyen el papel del analista, que es el responsable de que el producto final sea adecuado y de calidad.



También han hablado de las ventajas de usar herramientas de modelado, y de las aplicaciones de modelado de BD disponibles en el mercado, que son muchas más de las que en principio uno tiende a imaginar.

Todo esto le ha parecido a Víctor interesante, instructivo y fácil... así que le dice a Carmen que no le de más vueltas, que ya está listo para sentarse delante de alguna de esas herramientas CASE de las que tanto le ha hablado, para ponerse a trabajar. Carmen le dice que todo lleva su tiempo, y que aunque la herramienta que van a usar por ahora es sencilla de aprender, también requiere un periodo de aprendizaje.... Facilita el trabajo, pero no lo hace todo sola.

En lo que resta de esta unidad didáctica **aprenderás a utilizar herramientas CASE de modelado de datos** que te ayudarán en las tareas de diseño de bases de datos correctas. Comprobarás cómo **facilitan enormemente las labores de análisis**, permitiendo gestionar de forma sencilla, rápida y organizada las distintas fases del diseño. Es importante que recuerdes que **estas herramientas software no sustituyen el papel del analista, sino que le facilitan su trabajo**. Podemos afirmar que CASE supone "la informatización de la informática"



Empezaremos por conocer las **características comunes** que podemos encontrar en este tipo de aplicaciones y las ventajas que podemos encontrar en su utilización.

Posteriormente en esta unidad didáctica nos centraremos en una herramienta real, disponible en el mercado, para practicar los conceptos expuestos en unidades anteriores relativos al diseño E/R, modelo relacional y paso al sistema gestor de base de datos utilizando lenguaje SQL.

#### **Para saber más**

**Te presentamos una interesante monografía sobre herramientas CASE, que completa lo expuesto en esta unidad y abunda en los conceptos expuestos aquí.**

**[Monografía sobre herramientas CASE](#)** [\[Versión en caché\]](#)

Diseño de bases de datos relacionales

### **Características de las aplicaciones de modelado de BD**

**En el mercado existen multitud de herramientas CASE para modelado de datos**, cada una tiene sus propias peculiaridades, fruto de las decisiones tomadas por los equipos de desarrollo de las distintas empresas desarrolladoras y su manera de entender el modelado de datos, los detalles de implementación visual, las funciones incorporadas al sistema, su versatilidad, balance entre facilidad de uso y potencia, etc. Pero podemos encontrar características comunes en todas ellas.



**Se puede decir que todas comparten las siguientes características.**

- Implementación de las reglas de modelado según los modelos E/R y relacional.
- Gestión de un [diccionario de datos](#) donde se almacenan los elementos creados para el diseño



de la base de datos.

- **Comunicación con uno o más sistemas de gestión de base de datos para exportar de forma cómoda el modelo conceptual creado** en la herramienta a su implementación física, en ocasiones se puede realizar [ingeniería inversa](#) para importar el modelo conceptual a partir del esquema físico existente en la base de datos real.
- **Comprobación y optimización del modelo diseñado.**
- **Generación automática de la documentación** relativa al diseño realizado.
- **Interfaz de usuario cómoda y visual** para presentar de forma clara el diseño de la base de datos.

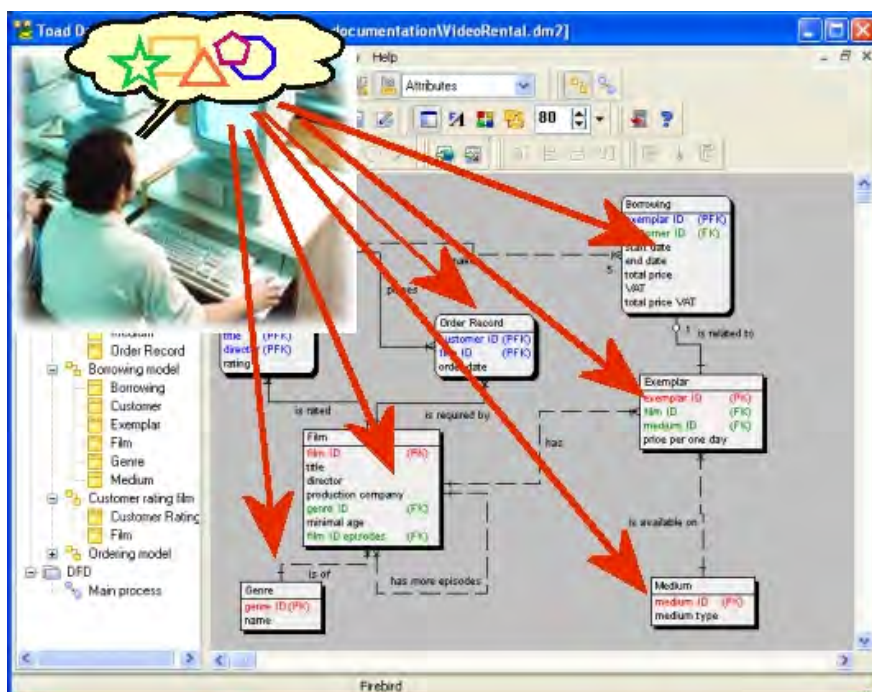
## Diseño de bases de datos relacionales

### Ventajas de usar herramientas de modelado de BD

De todo lo expuesto hasta ahora, ya habrás deducido **que la principal ventaja de utilizar estas herramientas** como apoyo a las labores de diseño de bases de datos **es conseguir la mayor productividad posible en el proceso, reduciendo los costes de esfuerzo y tiempo de diseño.** Este objetivo se consigue a través de diversos frentes.



- **Facilitan la aplicación práctica de metodologías de diseño estructuradas** (como el modelo E/R), para agilizar y sistematizar el trabajo de los diseñadores de aplicaciones.
- **Permiten el rápido desarrollo de prototipos de aplicaciones**, sobre todo en proyectos de gran envergadura.
- **Simplifican enormemente el mantenimiento** ordenado y estructurado de las aplicaciones, y en concreto de las bases de datos utilizadas.
- **Ayudan a los diseñadores en la producción y estandarización de la documentación** de las bases de datos.
- **Aumentan la portabilidad entre sistemas** de las bases de datos diseñadas.
- **Aseguran la reutilización de componentes software**, lo que redundará en una mayor eficiencia y la reducción de costes de desarrollo.
- **Representan una ayuda inestimable en la planificación de bases de datos** a lo largo del tiempo, sobre todo en los casos en que éstas son complejas.



### Autoevaluación

¿Qué no hace una herramienta CASE?

- a) Sustituye la labor de los analistas de aplicaciones.
- b) Generan automáticamente la documentación.
- c) Permiten realizar prototipos fácilmente.
- d) Simplifican el mantenimiento de aplicaciones diseñadas con ellas.

Comprobar

Diseño de bases de datos relacionales

### Aplicaciones de modelado de BD disponibles en el mercado

La oferta actual de aplicaciones informáticas disponibles en el mercado que pueden ser consideradas CASE es amplísima y también muy variada en cuanto a concepción, diseño y cantidad de funciones implementadas. Mientras unas sólo incluyen unas pocas capacidades CASE o se concentran en una sola fase del ciclo de vida del software, otras en cambio abarcan todas las fases, desde el diseño preliminar hasta el mantenimiento y explotación. Por supuesto esto también se refleja en el coste económico y de aprendizaje de las mismas.



Algunas de las aplicaciones CASE actuales son:

1. Xcase - <http://www.xcase.com/>
2. ArgoUML - <http://argouml.tigris.org/>
3. Blue Ink - <http://www.blueink.biz/>
4. CASE Studio 2 / TOAD Data Modeler - <http://www.casestudio.com/enu/default.aspx>
5. DBDesigner 4 - <http://fabforce.net/dbdesigner4/>
6. DMS Software Reengineering Toolkit - <http://www.semanticdesigns.com/Products/DMS/DMSToolkit.html>
7. Eclipse - <http://www.eclipse.org/>
8. Embarcadero ER/Studio - <http://www.embarcadero.com/products/erstudio/index.html>
9. Enterprise Architect - <http://www.sparxsystems.com.au/>
10. ERWin - <http://www3.ca.com/Solutions/Product.asp?ID=260>
11. Umbrello - <http://uml.sourceforge.net/>
12. iRise - <http://www.irise.com/>
13. IRqA - <http://www.irqaonline.com/>
14. Obsydian / Plex - <http://www3.ca.com/Solutions/Product.asp?ID=258>
15. Oracle Designer - <http://www.oracle.com/technology/products/designer/index.html>
16. Rational Rose - <http://www-306.ibm.com/software/rational/>
17. SILVERRUN - <http://www.silverrun.com/>
18. Sybase PowerDesigner - <http://www.sybase.com/products/developmentintegration/pc>
19. System Architect - <http://www.popkin.com/>
20. Together - <http://www.borland.com/together/>
21. Topcased - <http://www.topcased.org/>
22. Visible Enterprise Products - <http://www.visible.com/Products/index.htm>
23. Visual Paradigm for UML - <http://www.visual-paradigm.com/vpuml.php>
24. Rational ClearCASE - <http://www-306.ibm.com/software/rational/>
25. Database Designer for MySQL - <http://microolap.com/products/database/mysql-designer/>
26. swREUSER - <http://www.reusecompany.com/>
27. GeneXus - <http://www.genexus.com/>



Como ves la lista es muy extensa, pero en este momento nos interesan aquellas aplicaciones CASE que nos



ayuden en el diseño y modelado de bases de datos. En el siguiente apartado empezará a conocer en la práctica una de ellas, **TOAD Data Modeler**, que nos servirá como herramienta CASE para el diseño de nuestras bases de datos. No olvides que lo aprendido para esta aplicación será extensible a casi cualquier otra.

**Para saber más**

*Para que te hagas una idea de la envergadura del tema que estás estudiando, y compruebes la inmensa variedad de soluciones existentes en el mercado actual es muy interesante que visites el siguiente enlace que recopila enlaces a herramientas CASE clasificadas por categorías. Impresionante, ¿Verdad?*

[Listado de aplicaciones CASE clasificadas](#)

Diseño de bases de datos relacionales

**CASO.**

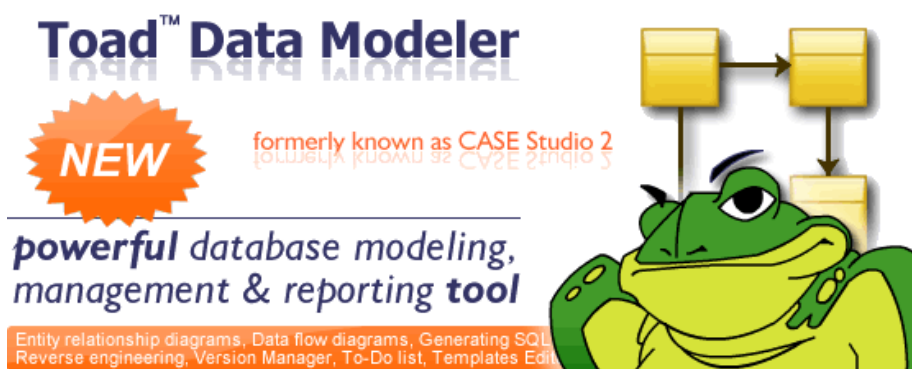
**Víctor** no deja de sorprender a **Carmen** por su facilidad para asimilar las cosas, sobre todo cuando se trata de hacerse con el manejo de una nueva herramienta. Él solito ha buscado en Internet, descargado e instalado la aplicación en un periquete, nada más darle **Carmen** el nombre: TOAD Data Modeler. Mientras **Carmen** atendía otras tareas, ha sido capaz, él solo, de identificar las posibilidades de esta herramienta, y compararla incluso con las características generales de las herramientas CASE que antes le había comentado **Carmen**. No obstante, como es lógico, todavía se pierde en los menús, y no sabe exactamente qué botones de las barras de tareas debe usar para hacer el trabajo que le interesa: Modelar su ejemplo de Base de datos.



**Carmen** le propone empezar con un ejemplo bastante más simple: Diseñar un diagrama E/R con dos entidades relacionadas, por ejemplo la relación entre sucursales de un banco y los empleados del banco, sobre la que ya habían trabajado antes, y que por tanto es conocida para **Víctor**.



Después de tanto oír hablar de las herramientas CASE estarás deseando empezar a practicar con una de ellas. Bien, ese momento ha llegado. **En el resto de la unidad didáctica aprenderás a utilizar la aplicación TOAD Data Modeler** de la empresa Quest Software. Comprobarás qué útil puede llegar a ser para diseñar bases de datos aplicando lo estudiado en las unidades anteriores (**modelos E/R y relacional, normalización y SQL**).



Hemos elegido TOAD Data Modeler como herramienta didáctica por tres razones:

- Es muy potente y dispone de muchas de las características mencionadas anteriormente.
- Soporta una gran cantidad de bases de datos.
- Dispone de una versión gratuita.

Aunque aquí nos centraremos en TOAD Data Modeler, te recomendamos que descargues alguna otra aplicación de la lista que te suministramos anteriormente y la compares en cuanto a

funcionalidad, posibilidades, facilidad de uso, etc. con la que aquí te mostramos. **Por ejemplo, te recomendamos MySQL Workbench que está totalmente orientada a MySQL**, el sistema gestor de base de datos utilizado en la unidad didáctica dedicada a SQL.

#### **Zona de descarga**

*Puedes descargar la herramienta MySQL Workbench de la página web de MySQL. Suponemos que a estas alturas conoces bien esta página, que por otra parte cambia con frecuencia, y puedes encontrar por ti mismo el enlace para descargar esa herramienta. Sólo intentamos facilitarte el trabajo con un enlace directo, y te hacemos notar que ahora se distribuye en dos paquetes. Por un lado podemos descargar el Workbench:*

- **MySQL Workbench**

*y el otro paquete a descargar incluye herramientas de interfaz gráfica para MySQL, con incluye los siguientes productos:*

- **MySQL Administrator**
- **MySQL Query Browser**
- **MySQL Migration Toolkit**

**Descarga de MySQL Workbench, junto al resto de herramientas GUI para MySQL**

<http://dev.mysql.com/downloads/gui-tools/5.0.html>

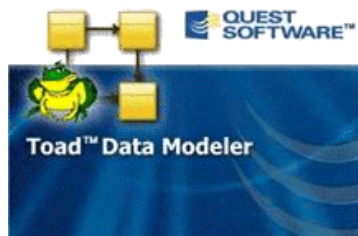
<http://dev.mysql.com/downloads/gui-tools/5.0.html>

Y sin más preámbulos conozcamos ya a **TOAD Data Modeler**...

---

Diseño de bases de datos relacionales

#### **Descarga e instalación de TOAD Data Modeler**



**TOAD Data Modeler es una aplicación disponible para descarga a través de Internet**, y podemos obtenerla fácilmente desde su página Web. **Existen dos versiones** de esta herramienta CASE, la versión **comercial** y la versión **gratuita**. **La mayor diferencia entre las dos es que la opción de ingeniería inversa sólo está disponible para la versión comercial**. Nosotros utilizaremos la versión gratuita.

#### **Zona de descarga**

*En el siguiente enlace puedes descargarte la herramienta Toad Data Modeler.*

[Enlace para descargar la herramienta TOAD Data Modeler.](#)

Resulta muy conveniente que te detengas en los diferentes enlaces de la página de TOAD Data Modeler. Está en inglés, como suele ocurrir en este tipo de productos, pero no es difícil de entender. Además, como en otras ocasiones, te animamos a practicar con este idioma, ya que es la lengua de la informática, y es importante para tu formación como profesional habituarte a trabajar con herramientas y documentación disponible en inglés.

**Una vez descargado el instalador de TOAD Data Modeler debes ejecutarlo e instalarlo.** Es un proceso muy sencillo, pero para que no tengas dudas te hemos preparado una animación con los detalles del proceso.





## Observa la instalación de TOAD Data Modeler

Si todo está correcto podemos empezar a trabajar con TOAD Data Modeler. Comencemos examinando sus posibilidades.

### Autoevaluación

Señala la respuesta correcta. TOAD Data Modeler es...

- a) Una herramienta CASE.
- b) Una aplicación informática.
- c) Una herramienta de modelado de bases de datos.
- d) Todas las respuestas anteriores son correctas.
- e) Ninguna de las respuestas anteriores es correcta.

Comprobar

---

Diseño de bases de datos relacionales

### Posibilidades de TOAD Data Modeler

Seguramente ya has visitado los diferentes enlaces de la página web de TOAD Data Modeler, y habrás comprobado que existe una tabla comparativa con **las posibilidades de la versión gratuita y la comercial**. A continuación te resaltamos las más importantes.



- **Soporte para más de 40 sistemas gestores de bases de datos** relacionales del mercado actual.
- **Posibilidad de generar informes de documentación de diseño en varios formatos:** HTML, RTF, JPG, BMP, PNG, XML.
- **Creación de diagramas E/R**, usando los elementos propios de este modelo de representación: entidades, atributos, dominios, claves primarias, claves ajenas y relaciones.
- **Definición del modelo relacional y físico** usando tablas, columnas, tipos de dato, índices, restricciones, etc. Todo ello adaptado al sistema gestor de bases de datos que estemos utilizando.
- **Generación de scripts SQL adaptados al sistema gestor de bases de datos deseado** para trasladar el diseño definido desde TOAD Data Modeler. Incluso es posible realizar cómodamente conversiones entre sistemas gestores.
- **Facilidades de configuración y ajustes de presentación** visual del modelo diseñado.
- **La versión gratuita está limitada a 25 entidades por diagrama.**

---

Diseño de bases de datos relacionales

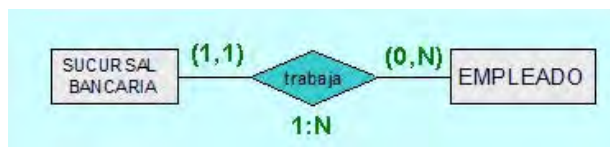
### Practicando el diseño de BD con TOAD Data Modeler

**Empezaremos nuestro trabajo con TOAD Data Modeler de la manera más sencilla.** Diseñaremos un diagrama E/R con dos entidades relacionadas. No te preocupes si no entramos en todos los detalles de la aplicación en este momento, profundizaremos en ellos más adelante. Ahora lo importante es que te familiarices con el interface de usuario y la operativa de la aplicación.

**Vamos a crear un diagrama E/R que represente una relación entre dos entidades, por ejemplo la relación entre sucursales de un banco y los empleados del banco.**



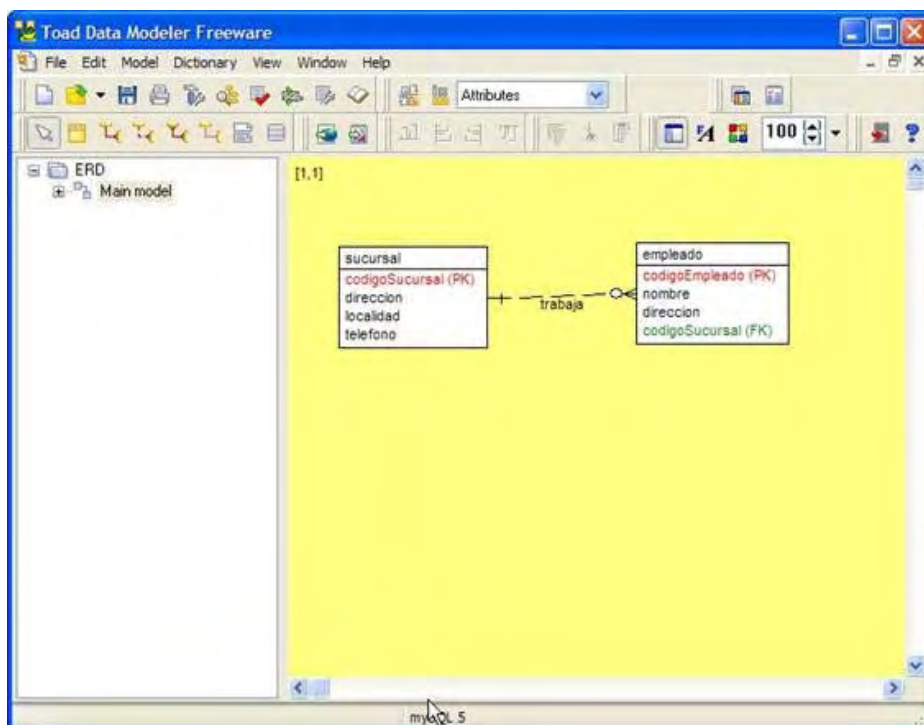
En este caso se trata de una relación binaria entre dos entidades, es obvio que la cardinalidad de la relación es de 1 a muchos (1:N), puesto que en una sucursal pueden trabajar varios empleados, pero un empleado sólo trabaja en una sucursal. En cuanto a la cardinalidad de las entidades, un empleado trabaja en una y sólo una sucursal. En cambio en una sucursal pueden trabajar 0 empleados (sucursales nuevas) o varios.



Hay que hacer notar que **TOAD Data Modeler no emplea el sistema de notación E/R "puro"** empleado en las unidades anteriores, y **las relaciones no son representadas como un rombo**. En el caso de las relaciones 1:N se efectúa automáticamente la propagación de claves, tal como se estudió anteriormente en la unidad número 3. En realidad **TOAD Data Modeler mezcla conceptos del modelo E/R con conceptos del modelo lógico relacional buscando una mayor productividad y aplicación práctica**, esto es bastante común en las herramientas CASE.



Así, en TOAD Data Modeler, el diagrama E/R anterior quedaría de la siguiente manera:



**Puedes ver cómo hemos llegado al resultado anterior siguiendo la animación que hemos preparado.**

También puedes descargarte el resultado final, en formato Toad Data Modeler, para que puedas verlo con esta herramienta CASE, ya instalada en tu equipo.

[!\[\]\(6059a5aa8b4ca7bb793408023d6c6e42\_img.jpg\) Descarga el resultado final](#)

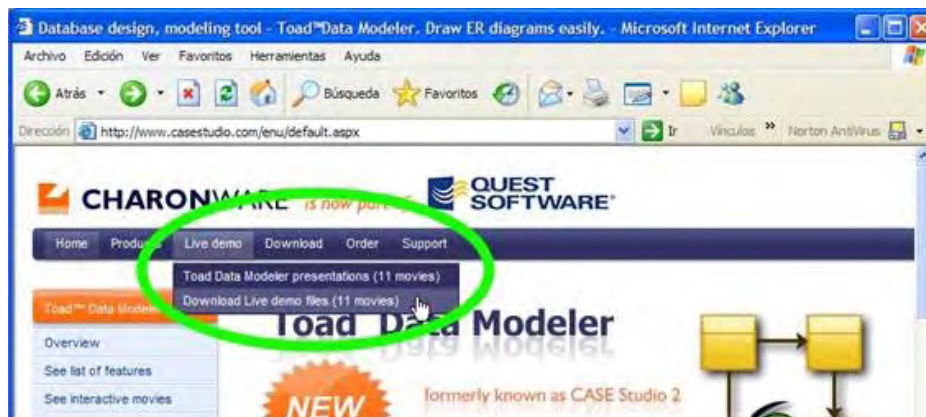
Ya habrás comprobado la utilidad de utilizar una herramienta CASE en nuestros diseños de bases de datos, y sobre todo en diseños complejos con muchas entidades y relaciones, en los cuales el método de "lápiz y papel" sería muy engorroso y poco productivo.

**Para saber más**

**Para que te familiarices aún más con TOAD Data Modeler y compruebes sus posibilidades, te pedimos que visualices detenidamente las animaciones que la**

empresa que lo desarrolla ha incluido en su página web. Resultan muy interesantes y aprenderás mucho sobre lo que se puede hacer y cómo hacerlo.

Animaciones mostrando las posibilidades de TOAD Data Modeler y su operativa básica



Diseño de bases de datos relacionales

#### CASO.

La primera impresión que **Víctor** ha tenido de esta herramienta ha sido muy buena. Al principio pensaba que quizás el hecho de que estuviera en inglés le podría plantear algún problema, pero se ha dado cuenta de que es bastante intuitiva, y que ya que está habituado a usar otras aplicaciones en inglés, y al final el hecho de no tener un gran nivel de este idioma, tampoco ha sido ningún obstáculo.



En cualquier caso, **Víctor** ya se ha hecho con el funcionamiento básico de esta herramienta, pero sabe que para comenzar a hacer el modelado de su problema (Departamentos, Empleados, Proyectos) le falta mucho que aprender.

Como en este proyecto tienen algo de prisa, y quieren que **Víctor** se encargue de él, porque **María** y **Carmen** están bastante ocupadas con otros proyectos, han decidido que el tiempo de aprendizaje se debe reducir al máximo, y a ser posible rentabilizarlo haciendo que al mismo tiempo que **Víctor** va aprendiendo a usar TOAD Data Modeler, vayan desarrollando algunos aspectos del proyecto. Por eso, **José**, que es el que más dominio tiene con esta herramienta, va a ser el encargado de explicarle su funcionamiento al mismo tiempo que entre ambos van desarrollando el modelo para su problema.

Lo primero que han tenido que elegir es el SGBD que van a usar finalmente para alojar la BD, y se han decidido, como en otras ocasiones, por MySQL. Seguidamente, **José** le ha explicado la conveniencia de elaborar un Diccionario de Datos con esta herramienta, y las ventajas que eso supone, en claridad y facilidad de modificación. Una vez hecho, definen las entidades los atributos, así como las relaciones existentes entre entidades, todo ello de una forma muy gráfica e intuitiva. Llegados a este punto, ya tienen construido el esquema relacional, tal y como lo habrían hecho "a mano", pero de una forma mucho más cómoda y sobre todo, mucho más rápida, al mismo tiempo que la herramienta hace comprobaciones que eviten algunos errores o diseños deficientes.



**Víctor** cree que ya han terminado, pero pronto se da cuenta de que está muy equivocado. **José** le muestra cómo a partir de ese esquema que han construido, sin hacer nada más, más que seleccionar un botón y completar algunos datos en un formulario, esta herramienta CASE genera una completa documentación del diseño de la base de datos, además, permitiendo elegir si esa documentación la queremos en formato HTML, lista para incluir en una

*página web, o bien en formato RTF, para poder verla y usarla un procesador de textos. Esto, a Víctor le ha parecido fantástico, pero cuando ve que además, de forma también automática, la herramienta CASE genera un Script SQL con todas las sentencias necesarias para crear de forma efectiva esa Base de Datos en el SGBD, sencillamente se queda sin palabras. ¡La de trabajo que se ha ahorrado escribiendo sentencias SQL para el SGBD! Ahora bastará con ejecutar el Script SQL desde el propio SGBD, y nuestra base de datos estará lista para usar.*



En este momento ya tienes una idea formada de qué es, para qué sirve y cómo se utiliza una herramienta CASE de modelado de base de datos. Ahora **vamos a desmenuzar las posibilidades que nos ofrece**, tratándolas de una manera separada.

**Como hilo conductor de nuestro aprendizaje utilizaremos un ejemplo sencillo que nos permitirá trabajar de principio a fin las diferentes funciones incorporadas en la aplicación de modelado de bases de datos.** Es importante que sigas los siguientes apartados, realizando por ti mismo en tu ordenador las operaciones indicadas.

## Diseño de bases de datos relacionales

### Un ejemplo para practicar

Como la mejor manera de aprender a hacer las cosas es haciéndolas, **vamos a tomar un ejemplo sencillo y empezar a practicar con TOAD Data Modeler.** Para que el ejemplo no sea un problema y puedas concentrarte en la herramienta en sí y su **manejo hemos escogido un ejemplo que debes conocer muy bien**, puesto que se utilizó intensivamente en la unidad dedicada al estudio de SQL. Nos estamos refiriendo al ejemplo que trataba sobre el siguiente supuesto:

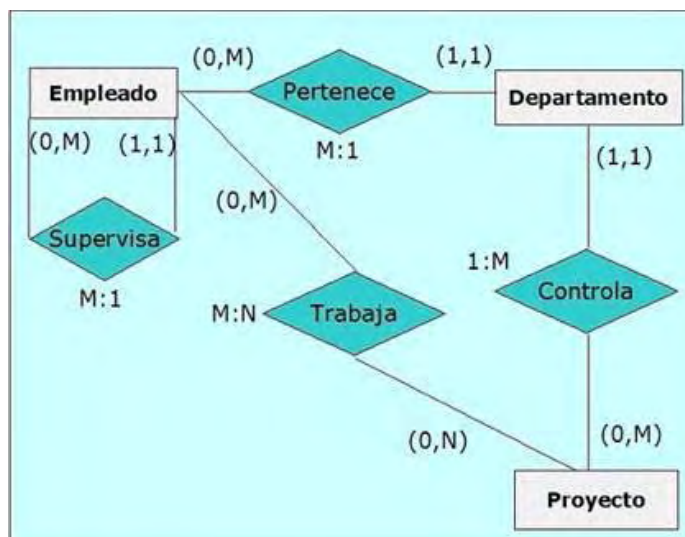
Una **empresa** pretende desarrollar una base de datos de **empleados y proyectos**.

- La empresa está estructurada en **departamentos**, en cada uno de los cuales se están desarrollando uno o varios **proyectos**, de forma que **un proyecto sólo depende de un departamento**.
- Por otro lado, **cada departamento consta de uno o varios empleados** que trabajan de forma exclusiva para ese departamento, **pero pueden trabajar simultáneamente en varios proyectos**.
- **Cada empleado tiene un jefe** encargado de supervisar su trabajo, pudiendo cada jefe supervisar el trabajo de varios empleados.
- **Tanto los empleados, como los departamentos y los proyectos se identifican por un código que es único** para cada uno de ellos.
- De cada empleado se quiere guardar su nombre y su fecha de ingreso en la empresa, así mismo todos los departamentos y los proyectos tienen un nombre descriptivo.
- Por último, **se quiere guardar en la base de datos el número de horas que ha trabajado cada empleado en los proyectos en los que está implicado.**



**¿Recuerdas el ejemplo? Pertenece a la unidad 5 y en aquel momento llegamos al siguiente diagrama E/R:**





Como ya estamos muy familiarizados con este ejemplo, su comprensión no será un obstáculo y esto nos permitirá concentrarnos en el aprendizaje de la herramienta CASE.

Preparados, listos... ¡Manos a la obra!

Diseño de bases de datos relacionales

### Elección del Sistema Gestor de Bases de Datos



Ya sabes que el fin último de todo modelo E/R que diseñemos es ser trasladado a un sistema gestor de bases de datos relacionales. Una de las formas de conseguirlo es transformar el diagrama E/R en un conjunto de sentencias SQL. TOAD Data Modeler utiliza este sistema. Por otra parte también conoces que aunque SQL es un lenguaje estandar, cada sistema gestor de bases de datos lo dota de particularidades propias, o bien no implementa el estandar en su totalidad.

Por todo esto **TOAD Data Modeler debe conocer de antemano el sistema gestor de bases de datos que estamos utilizando**, para generar un script SQL adecuado para él. Así, cuando creamos un nuevo modelo lo primero que se nos requiere es el gestor de bases de datos con el que estamos trabajando.



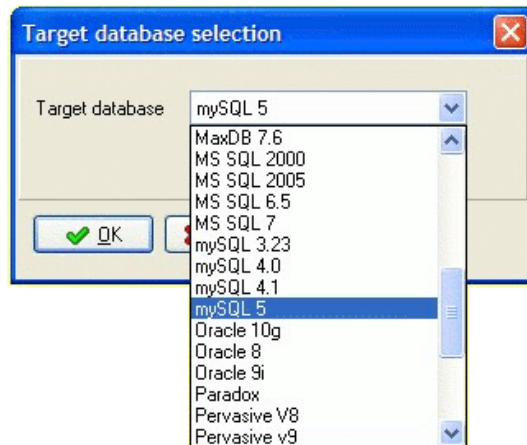
TOAD Data Modeler, tiene soporte para más de 40 sistemas gestores de bases de datos presentes en el mercado actual, así que podemos decir que es una herramienta bastante versátil.

TOAD Data Modeler se adapta a nuestra elección, de forma que los menús y opciones que se presentarán a partir de ese momento estarán mediatizados por nuestra elección.

**En esta unidad utilizaremos MySQL 5 como gestor de bases de datos puesto que ya lo conoces de la unidad anterior.** En unidades posteriores comprobarás las posibilidades de pasar de un sistema a otro que ofrece TOAD Data Modeler, puesto que introduciremos para su estudio y aprendizaje otro sistema gestor de bases de datos muy utilizado en la actualidad, **Oracle**.



La elección del sistema gestor de base de datos se hace antes de empezar a diseñar el diagrama E/R. Sólo hay que seleccionar el sistema adecuado de la lista desplegable que se nos presenta.



## Autoevaluación

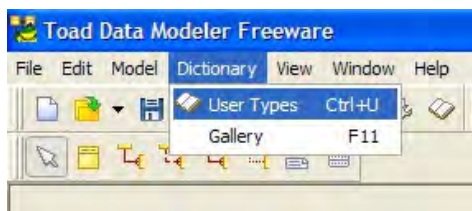
TOAD Data Modeler permite exportar el diseño realizado a....:

- a) MySQL 5.
- b) Oracle.
- c) Un script SQL.
- d) Ninguna es correcta.

Comprobar

Diseño de bases de datos relacionales

## Diccionario de datos



**Para conseguir un diseño organizado y fácilmente actualizable de nuestra base de datos es muy útil usar el diccionario de datos incluido en TOAD Data Modeler.** El diccionario de datos nos permite crear y mantener nuestros propios tipos de datos, mejorando mucho la productividad de nuestro trabajo al mismo tiempo que se consigue una mayor facilidad de actualización del diseño y se previenen errores.

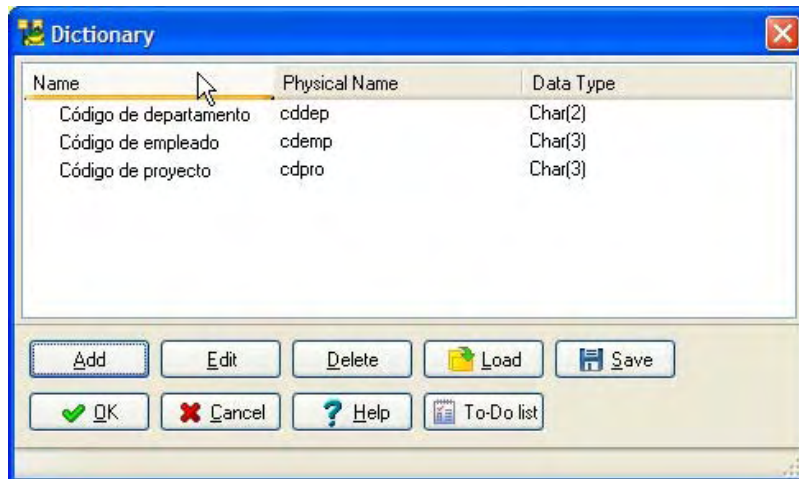
Por **ejemplo**, si definimos el tipo de dato de usuario "codigoEmpleado" como CHAR(3) cuando creamos un atributo de una entidad podremos especificar como tipo de dato "codigoEmpleado" en lugar de CHAR(3). Esto tiene varias ventajas:

- **Mejora la legibilidad.**
- **Facilita el mantenimiento y los cambios en el diseño.** Si utilizamos el tipo de datos definido en varias entidades y en un momento dado decidimos cambiar de CHAR(3) a CHAR(20) sólo tendremos que cambiar en el tipo de dato de usuario y el cambio se reflejará en todos los atributos de todas las entidades que lo utilicen.
- **La documentación generada será más completa.**



**En la siguiente animación te indicamos el proceso para crear un tipo de dato definido por el usuario.**

Después de crear los tipos de dato de usuario de nuestro ejemplo el diccionario quedaría así:



Observa cómo **podemos mantener nombres diferentes para los tipos de dato a nivel de diagrama E/R y a nivel de columnas de tablas SQL** (Name y Physical Name respectivamente). **Esto permite que los diagramas sean más legibles** puesto que se pueden incluir tildes y espacios en blanco. Aunque **no hay que olvidar que en el fondo representan lo mismo**.

**También es interesante resaltar que no es necesario hacer un tipo de dato para cada posible atributo de nuestro modelo.** En general deberíamos crear tipos de dato de usuario en aquellos casos que sea posible la **reutilización**.

También es posible crear un diccionario de entidades que pueden ser reutilizadas en diferentes diagramas E/R, para ello se utiliza la galería (Gallery).

### Autoevaluación

Señala las respuestas correctas.

- a) Deberíamos crear una entrada en el diccionario de datos para cada clave primaria.
- b) Los atributos siempre deben basarse en una entrada del diccionario.
- c) Una vez que se define un tipo de dato no puede cambiarse.
- d) Ninguna es correcta.

Comprobar

Diseño de bases de datos relacionales

### Entidades y atributos

**Las entidades, junto con sus atributos, y las relaciones, que se tratarán en el siguiente apartado, son los elementos que nos permiten diseñar nuestros diagramas E/R.** Veamos a través de nuestro ejemplo como se utilizan en TOAD Data Modeler.

**Añadir una entidad es tan fácil como pulsar el botón de entidad** (Entity) y colocarla en el lugar deseado. Después podemos editarla haciendo doble click sobre ella. Aparecerá la ventana con las propiedades de la entidad agrupadas en forma de pestañas.





La pestaña más importante es la de atributos (Attributes), ya que en ella se definen los atributos de la entidad y todas sus propiedades:

- **Clave**, (Key): indica si el atributo forma parte de la clave primaria, de la ajena, de ambas o de ninguna.
- **Nombre del atributo en la entidad**, (Name).
- **Nombre de columna que tendrá en la tabla de SQL**, (Column Name).
- **Tipo de dato**, (Datatype): puede ser un tipo de dato SQL (por ejemplo CHAR, DATE, INTEGER, etc) o un tipo de dato definido por nosotros.
- **Not null**: indica si el atributo puede o no tomar valores nulos.
- **Unique**: indica si el atributo puede o no repetirse en dos instancias de la entidad.
- **Descripción**, (Description): permite indicar información para mejorar la documentación.



Para que entiendas perfectamente el sentido de cada una de las pestañas, nada mejor que practicarlo con un ejemplo. Tomemos la **entidad empleado** y veamos con una animación cómo llegar a la siguiente definición de atributos:

Empleado				
Atributo	Clave primaria	Clave ajena	Tipo de dato	
cdemp	Sí	No	CHAR(3)	
nombre	No	No	CHAR(30)	
fecha_ingreso	No	No	DATE	
cdjefe	No	Sí	CHAR(3)	
cddep	No	Sí	CHAR(2)	

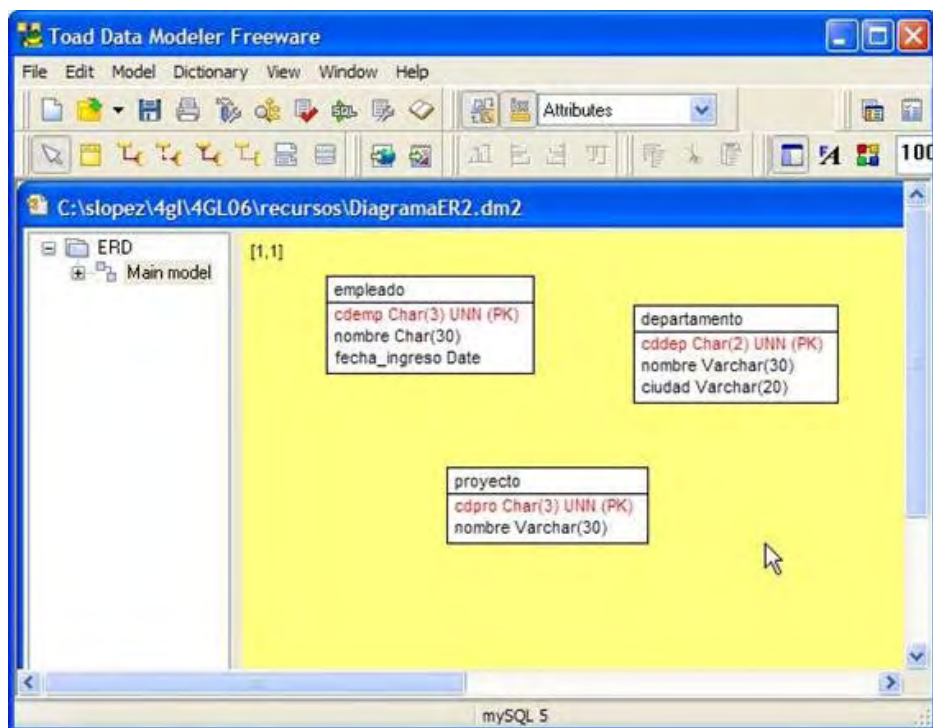
Hay que hacer notar que **en TOAD Data Modeler no hace falta definir los atributos que sean claves ajenas, ya que estos se generarán automáticamente al definir las relaciones entre entidades (propagación de claves)**. Esto lo veremos en el siguiente apartado. Por lo tanto en el caso de la entidad empleado sólo debemos definir los atributos: cdemp, nombre y fecha\_ingreso.



#### Definición de entidades

Ahora puedes practicar creando las otras entidades de nuestro ejemplo: **departamento y proyecto**. Deberías obtener el siguiente esquema:





Diseño de bases de datos relacionales

## Relaciones




¿Vas familiarizándote con la operativa de **TOAD Data Modeler**, verdad?

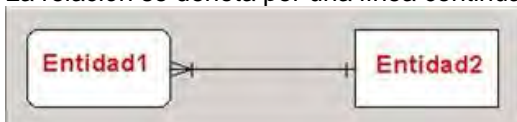
Como ves, si has ido practicando los ejemplos propuestos, **es una aplicación sencilla e intuitiva de usar, sobre todo si tienes claros los conceptos teóricos que se explicaron en las unidades anteriores sobre modelo E/R, modelo relacional, normalización y SQL.**


En este apartado vamos introducir uno de los elementos, junto con las entidades, que forman la base del modelo E/R, en el cuál se basa TOAD Data Modeler.

Veamos las **relaciones entre entidades**.

**En el modelo E/R las entidades siempre están unidas unas con otras a través de relaciones. TOAD Data Modeler maneja 5 tipos de relación entre entidades:**

 **Relaciones identificativas (Identifying relationship): Relacionan una entidad fuerte con otra débil y con cardinalidad 1 a N**, de manera que la clave ajena en la entidad débil pasa a formar parte de la clave primaria de esa entidad. La entidad débil se denota por un rectángulo con las esquinas redondeadas y la fuerte por un rectángulo con las esquinas rectas. La relación se denota por una línea continua. El diagrama quedaría de la siguiente forma:



 **Relaciones no-identificativas (Non-identifying relationship): Relacionan dos entidades con cardinalidad 1 a N**, de forma que la clave ajena en la entidad del lado N no forma parte de la clave primaria de esa entidad. La relación se denota por una línea discontinua. El diagrama queda como te mostramos a continuación:



 **Relaciones muchos a muchos (N:M): Relacionan dos entidades con cardinalidad N a**

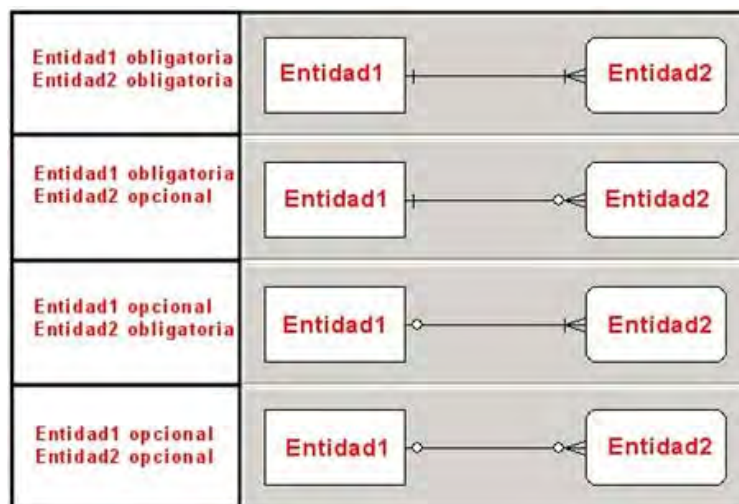
**M**, en este caso se crea automáticamente una nueva entidad (débil) cuya clave primaria es la conjunción de las claves primarias de las entidades relacionadas. A continuación se muestra un ejemplo de diagrama para este tipo de relaciones:



**Relaciones informativas** (Informative relationship): Son un tipo de relación que debe evitarse. **Relacionan dos entidades sin precisar su cardinalidad** y por tanto sin generar claves ajenas.

**Autorrelaciones:** Relacionan una entidad consigo misma, estas relaciones siempre son de cardinalidad 1 a N.

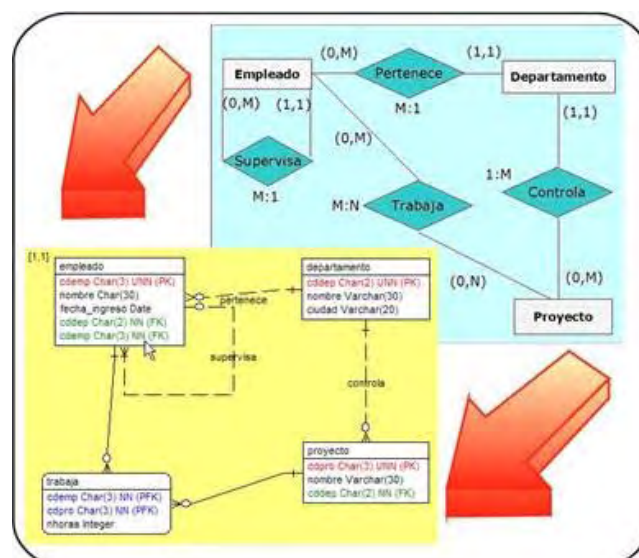
En cuanto a la **cardinalidad de las entidades** participantes en una relación, ya sabes que puede ser **obligatoria** (total) u **opcional** (parcial). En TOAD Data Modeler se denota de la siguiente forma:



Ya has comprobado que **toda la teoría del modelo E/R** que estudiaste en las unidades anteriores **está presente en los diagramas que pueden ser generados con TOAD Data Modeler**. Como la mejor manera de comprender los conceptos es utilizando un ejemplo, **vamos a aplicar lo estudiado sobre las relaciones a nuestro ejemplo de la base de datos de empleados, departamentos y proyectos**, y para hacértelo todavía más gráfico lo haremos con una animación.

### Definición de relaciones

Y aquí está nuestro ejemplo terminado. Compáralo con el esquema que utilizamos en la unidad 5 ¿Se parecen mucho, verdad?



## Autoevaluación

El número de tipos diferentes de relaciones en TOAD Data Modeler es:

- a) 4.
- b) 5.
- c) 6.
- d) 7.

Comprobar

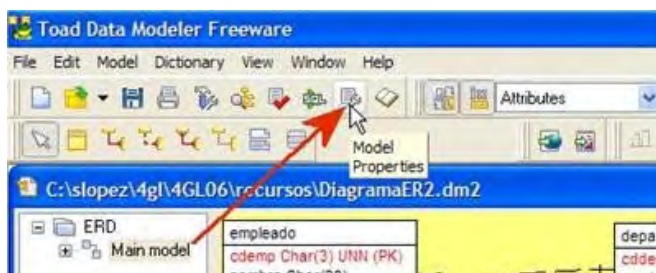
Diseño de bases de datos relacionales

### Generación automática de documentación

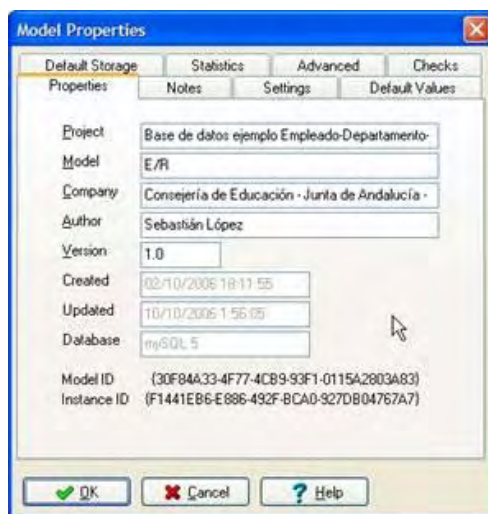
Vamos a comprobar ahora cómo **las herramientas CASE de diseño E/R**, y en concreto TOAD Data Modeler **nos ayudan a mantener una documentación ordenada, legible y, lo que es muy importante, sincronizada en todo momento con el diseño realizado.**

Lo primero que podemos hacer para conseguir una documentación completa es añadir a nuestro diseño un cuadro con las propiedades que va a tener el mismo. Esto se consigue fácilmente utilizando el botón de propiedades del modelo (**Model Properties**) y rellenando los campos que pone a nuestra disposición.

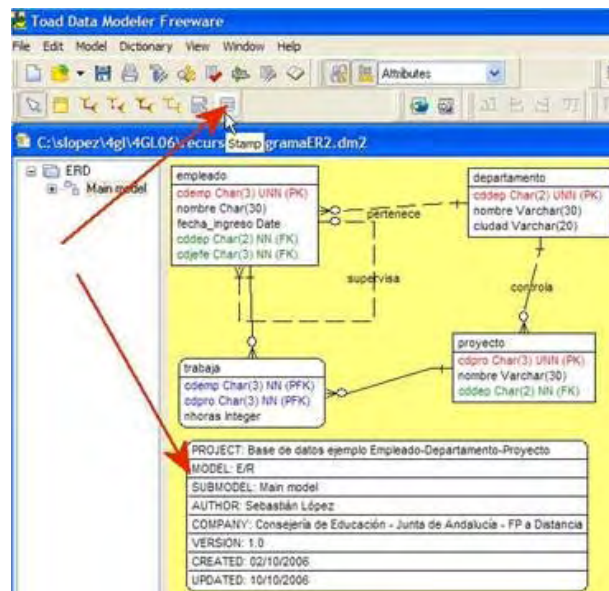
En la imagen siguiente se muestra la ubicación del botón de propiedades del modelo, dentro de la barra de herramientas de Toad Data Modeler.



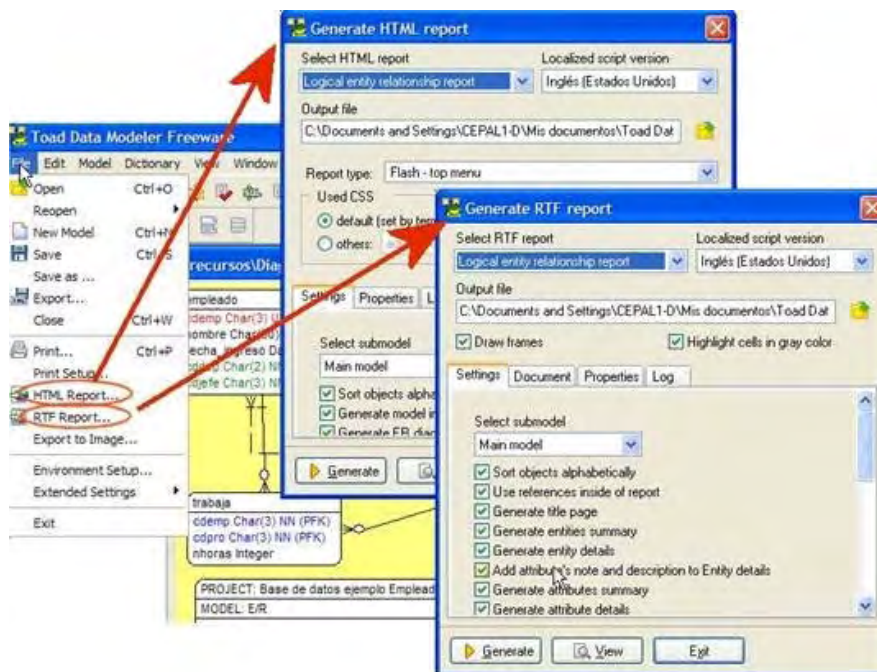
A continuación se muestra el aspecto del cuadro de diálogo que nos abre el botón de Propiedades del Modelo:



Finalmente podemos incluir esta información en nuestro diagrama por medio del botón **"Stamp"**, tal y como se muestra en la siguiente imagen:



Las posibilidades de autodocumentación de TOAD Data Modeler no acaban ahí, **ahora podemos generar toda la documentación relativa a nuestro diseño en formato HTML** (para verla en un navegador web) **o RTF** (para verla, por ejemplo en Microsoft Word).



**Te adjuntamos como recursos descargables las documentaciones generadas del modelo E/R que hemos venido utilizando como ejemplo.** Date cuenta de la potencia que nos ofrece la herramienta CASE y cómo eleva nuestra productividad

¿Cuánto tiempo tardarías en generar la misma documentación "a mano"? ¿Y si encima tuvieras que actualizarla cada vez que hicieses una modificación al diseño? Mejor no lo pienses... La respuesta es DEMASIADO.

[↓ Descarga la documentación](#)

[↓ Descarga la documentación comprimida](#)



## Generación de scripts SQL



Bien, estamos llegando al final. ¿Recuerdas cuál era nuestro objetivo final, en qué acababa todo?

Claro, **ahora hay que trasladar el diseño realizado a un sistema gestor de bases de datos relacionales**. Esto se consigue **generando desde TOAD Data Modeler un script SQL con el esquema de nuestra base de datos** en formato SQL adaptado al gestor utilizado. Ese script se lanzará desde el gestor de bases de datos, y... ¡Exacto! Tendremos nuestra base de

datos lista para ser usada.

Para generar el script SQL disponemos del botón generar script (**Generate Script**):



Podemos fijar muchas de las características del script. Fijaremos las que deseamos exportar al gestor de base de datos, y cuando estemos listos, generaremos un fichero de texto que contiene la definición de nuestra base de datos lista para ser cargada en el gestor de base de datos.



**Observa el proceso en la siguiente animación.**

---

Diseño de bases de datos relacionales

### Nuestro ejemplo terminado

Bueno, eso es todo, hemos **recorrido las funcionalidades más importantes de una herramienta CASE y las hemos aplicado a nuestro conocido ejemplo de la base de datos de gestión de proyectos**. Recuerda los pasos:

1. **Modelo E/R**
2. **Diseño en TOAD Data Modeler.**
3. **Generación de documentación.**
4. **Generación de script SQL para trasladar el modelo a un sistema gestor de bases de datos relacionales.**

Por supuesto ahora posiblemente vendría una fase de refinamiento del modelo que volvería a empezar por el primer paso en adelante.



**Imagina la cantidad de trabajo que las herramientas CASE pueden ahorrarnos, recuerda... ¡Productividad!**

Para terminar te facilitamos el modelo obtenido en los apartados anteriores en el formato de TOAD Data Modeler: Puedes abrirlo con la herramienta CASE en tu propio ordenador.

[!\[\]\(1f56542a42e2413e44a2b2023033aa2e\_img.jpg\) Descarga el modelo obtenido](#)

---

Diseño de bases de datos relacionales

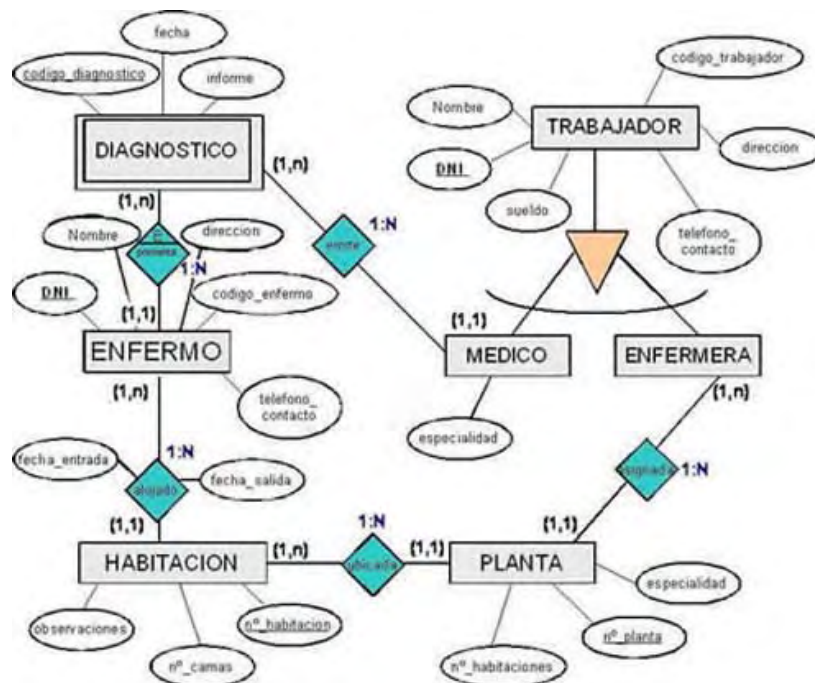


**Te proponemos ahora que practiques todo lo aprendido en esta unidad** diseñando con TOAD Data Modeler el modelo lógico de datos E/R de un problema que seguro recordarás, nos referimos al ejemplo del hospital que se trató en las unidades 2

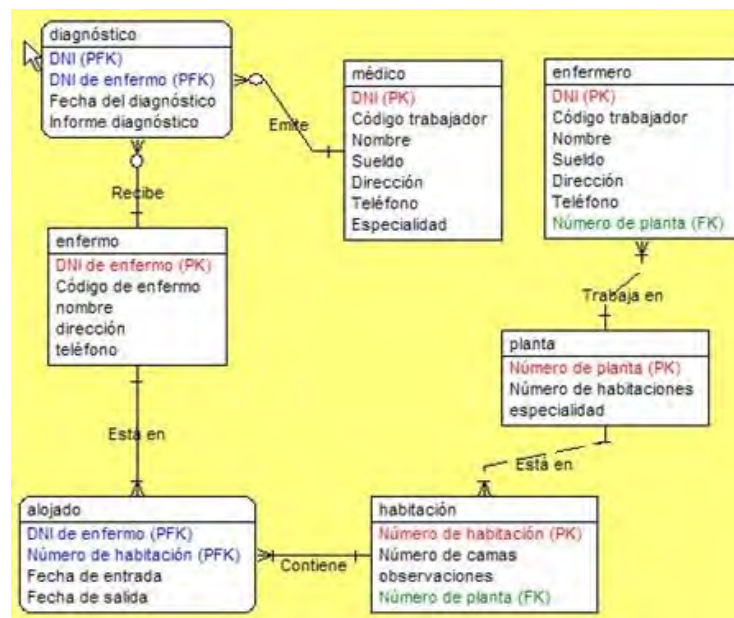
y 3.

Te recomendamos que releas los contenidos de esas unidades y recuerdes el planteamiento del problema, el esquema E/R que se propuso y su traslación a modelo relacional. Te pedimos que realices el esquema E/R con TOAD Data Modeler y generes la base de datos correspondiente en MySQL siguiendo las pautas marcadas en los apartados anteriores.

Debes reproducir en TOAD Data Modeler el siguiente esquema E/R:



Para llegar al siguiente resultado:



Para facilitarte el trabajo te proporcionamos el diseño en formato TOAD Data Modeler como un recurso que puedes descargar.

[📄 Descarga el diseño en formato TOAD](#)