

Programacion para la industria

Annie Stefania Medina

Jose Antonio Escobar

Luis Miguel Hoyos

July 2022

PARTE TEORICA

1 Modelamiento de datos a nivel conceptual

Como sugiere el nombre, el modelado de datos conceptual es más relevante en la etapa conceptual, cuando una empresa elabora un plan preliminar con la intención de trabajar en los detalles más finos más adelante. Generalmente creados por arquitectos de datos y partes interesadas comerciales, los modelos de datos conceptuales brindan a las partes interesadas una fotografía instantánea fácilmente digerible de los conceptos o entidades relevantes y las relaciones entre ellos. Al comunicar el modelo de una manera que sea relevante para las partes interesadas que no están necesariamente orientadas a la tecnología o los detalles, es más probable que los modeladores obtengan apoyo para sus proyectos.

El objetivo de un modelo de datos conceptual es proporcionar una perspectiva de la empresa centrada en los datos al documentar cómo se relacionan las diferentes entidades comerciales entre sí. Esto a menudo se logra a través de diagramas de relación de entidades (ERD) o modelos de funciones de objetos (ORM). A diferencia de los modelos de datos lógicos y físicos, los modelos de datos conceptuales son independientes de la tecnología y la aplicación. Esto significa que no están vinculados a la realidad y el contexto de los sistemas y procesos actualmente en vigor.

Los modelos de datos conceptuales demuestran tanto el estado actual como el futuro, lo que significa que incluyen cambios en el negocio que están ocurriendo o que son muy probables. De esta manera, las empresas pueden preparar el modelo para el futuro hasta cierto punto y tener en cuenta cualquier flexibilidad que deba incorporarse a la solución. Se considera una buena práctica distinguir entre el estado actual y el estado futuro mediante el esquema de colores.

¿Cuál es el objetivo de un modelo de datos conceptual?

Debe emplearse un modelo de datos conceptual para definir y comunicar relaciones de alto nivel entre conceptos/entidades. En otras palabras, ayudan a

una empresa a ver sus datos y las relaciones entre diferentes tipos de datos en contexto.

Idealmente, serán representaciones visuales de datos en contexto que cuenten la historia de cómo opera una empresa en circunstancias particulares. Esto puede ayudar a las empresas a evitar descuidos que podrían causar problemas importantes en el futuro. Por ejemplo, al crear o adquirir un nuevo sistema de gestión de relaciones con los clientes (CRM), es posible que no esté clara la necesidad de distinguir entre un cliente potencial y un cliente. Pero sin esa distinción, un cliente potencial "nuevo" podría ser en realidad un empleado de una empresa con una cuenta existente.

Una base de datos que reconoce la distinción entre un cliente potencial y un cliente también puede modelarse para reconocer cualquier relación potencial entre un nuevo cliente potencial y un cliente existente, al permitir que el registro se consolide. De esta manera, los representantes de ventas y los representantes de soporte tienen el contexto que necesitan para trabajar de manera eficaz.

¿Cuándo debería considerar un modelo de datos conceptual?

Los modelos de datos conceptuales se utilizan en las primeras etapas del modelado de datos para organizar y definir conceptos y reglas en función de los requisitos de los casos de uso. Son los menos detallados de los tres tipos de modelos de datos, pero de ninguna manera esto hace que sean menos útiles. De hecho, uno de los beneficios clave de los modelos de datos conceptuales es que se pueden comprender y comunicar rápidamente a las partes interesadas fuera de la burbuja "tecnológica".

Los modelos de datos conceptuales brindan a las empresas un punto de partida que debe evolucionar hacia diagramas más ricos en contexto a medida que avanzan por las etapas de los modelos de datos. A través del análisis de casos de uso, el diseño de casos de uso y el diseño de bases de datos, la complejidad y el nivel de detalle eventualmente alcanzarán su punto máximo con los modelos de datos físicos.

¿Por qué debería utilizar un modelo de datos conceptual?

Pasar por alto la etapa de modelado de datos conceptual aumenta la probabilidad de que se pierdan las relaciones entre entidades de gran tamaño, como la distinción cliente/cliente potencial mencionada anteriormente. Además, los modelos de datos conceptuales tienen en cuenta la cardinalidad. La cardinalidad describe cuáles pueden ser las relaciones de una entidad con otras entidades, incluidas de una con una, una con muchas o muchas con muchas.

Sin un modelo de datos conceptual, cuanto a mayor profundidad se encuentre la organización en el ciclo de desarrollo, los descuidos se vuelven más probables. Esto se debe a que el panorama general a menudo se pierde cuando los equipos se concentran en los detalles del diseño y están bajo la presión de los plazos. Al construir primero el modelo de base de datos conceptual, las empresas pueden

evitar tales descuidos y ver la terminología vaga, por lo que se considera y define con anticipación el potencial de relaciones entre entidades.

Beneficios de los modelos de datos conceptuales

Algunos de los beneficios clave del modelado de datos conceptual incluyen los siguientes:

- Ayudan a definir un lenguaje común y completar el glosario empresarial. Al proporcionar una fotografía instantánea de los conceptos, las entidades y sus relaciones, se pueden identificar las lagunas en la terminología y limitar los descuidos.
- Ayudan a dar forma a la hoja de ruta y definir el alcance del proyecto. Un modelo conceptual puede ayudar a las partes interesadas relevantes a comprender mejor lo que se requiere para lograr los resultados comerciales deseados desde una perspectiva de gestión de recursos y tiempo.
- Proporcionan una base para futuros modelos. El modelado de datos conceptual puede verse como el primer paso hacia tipos más profundos de modelado de datos. Una vez que se construye el modelo conceptual, los modeladores pueden comenzar a introducir más contexto en el modelo. Para un modelo de datos lógicos, esos serían los requisitos de datos de la base de datos. Para el modelado de datos físico, el contexto ahora incluiría el sistema de administración de bases de datos (DBMS) específico para el que se modela.
- Fomentan una forma de comunicación inclusiva. Garantizar una implementación exitosa requiere la participación de las partes interesadas dentro y fuera de la burbuja tecnológica. Los modelos de datos conceptuales ayudan a fomentar esta colaboración.

2 Modelado de datos lógico

El modelado de datos lógico es uno de los tres tipos, o etapas, del modelado de datos, junto con el modelado de datos conceptual y físico. A veces denominado modelado de información, el modelado de datos lógico es la segunda de estas etapas. Ayuda a las empresas a desarrollar una comprensión visual de la información que deben procesar para completar con éxito tareas o procesos comerciales específicos.

¿Qué es un modelo de datos lógico?

Como representación gráfica de los requisitos de información para un área comercial determinada, se construye un modelo de datos lógico tomando las descripciones de datos representadas en un modelo de datos conceptual e introduciendo

elementos asociados, definiciones y un contexto más amplio para la estructura de los datos.

Esta etapa es importante porque, si bien el modelo de datos conceptual más simplificado se comunica más fácilmente, la falta de contexto puede dificultar el paso del modelado a la implementación. Se requieren más detalles para respaldar esa progresión. Dichos detalles incluyen la definición de los atributos propios, las claves principales, las claves externas, la cardinalidad de las relaciones y la descripción de entidades y clases. En esta etapa, se establece y define la naturaleza de las relaciones entre los datos, y se normalizan los datos de diferentes sistemas.

¿Cuál es el objetivo de un modelo de datos lógico?

En esta etapa, la función principal del modelo de datos es visualizar los elementos de datos y cómo se relacionan entre sí. El modelado de datos lógico también funciona para detallar los atributos asociados con un elemento de datos. Por ejemplo, un modelo de datos lógicos especificaría la naturaleza de un elemento de datos, es decir, nombre de cuenta (cadena de caracteres), número de cuenta (entero).

¿Cuándo debería considerar un modelo de datos lógico?

Los tres tipos diferentes de modelos de datos proporcionan grados crecientes de contexto y detalle, por lo que podemos ver su uso de forma secuencial. Por lo tanto, se debe considerar un modelo de datos lógico una vez que se haya construido el modelo de datos conceptual.

Esta etapa más estructurada del modelado de datos es más relevante durante el diseño de la aplicación, cuando puede servir como mecanismo de comunicación en los entornos más técnicos donde trabajan los analistas y diseñadores de bases de datos. Nos ayuda a comprender los detalles de los datos en mayor medida que los modelos de datos conceptuales, pero tampoco llega a proporcionar una perspectiva sobre cómo se deben implementar.

Al igual que con el modelado de datos conceptual, esto significa que los equipos no están sujetos a consideraciones tecnológicas. Esto resulta importante, ya que la naturaleza de la tecnología en las empresas suele ser dinámica.

¿Por qué debería usar un modelo de datos lógico?

Una manera fácil de comprender cuándo y por qué un modelo de datos lógico sería relevante es considerar la audiencia a la que se dirige el modelo: analistas de bases de datos, analistas de sistemas y diseñadores. La audiencia y el lugar del modelado de datos lógicos en el proceso de diseño de aplicaciones significa que excluir el contexto y los detalles a favor de la accesibilidad es menos relevante. La capa adicional de detalle (en comparación con el modelado de datos conceptual) es el contexto que los arquitectos necesitan para garantizar que las

nuevas aplicaciones sean compatibles con los datos que abarcarán. Esencialmente, un modelo de datos lógicos proporciona los cimientos necesarios para un diseño productivo de bases de datos.

Sin un modelo de datos lógico, los diseñadores solo pueden descubrir realmente los requisitos de una nueva aplicación sobre la marcha. Esto a menudo significará trabajar con elementos de datos desorganizados, lo que hace que sea más probable pasar por alto dichos requisitos. Por lo tanto, omitir la etapa de modelado de datos lógico en favor de la construcción de un modelo de datos físico puede generar un diseño de base de datos deficiente y aplicaciones que no funcionen como se esperaba. Abordar estos errores requiere un enfoque reactivo que puede ralentizar el tiempo de comercialización y aumentar los costos totales asociados con el proceso de desarrollo.

Además, la naturaleza independiente de la tecnología de un modelo de datos lógico ayuda a las empresas a establecer oportunidades para mejorar los procesos. Esto significa que se pueden desarrollar nuevas aplicaciones para que sean lo más efectivas posible, en lugar de tan efectivas como lo permiten las limitaciones tecnológicas actuales.

Beneficios del modelado de datos lógico

Contexto, contexto, contexto. Cada etapa del proceso de modelado de datos beneficia a una empresa al introducir una nueva capa de contexto al modelo. Esto ayuda a desarrollar una imagen mucho más clara del estado actual de los sistemas y procesos de una empresa, lo que arroja luz sobre la mejor manera de navegar hacia el estado futuro deseado. Los beneficios específicos del modelado de datos lógico son los siguientes:

- Ayudan a las empresas a identificar áreas para mejorar los procesos comerciales. Al construir un modelo sin ataduras a las limitaciones tecnológicas actuales, las empresas pueden identificar lo que se requiere para concretar la versión ideal del modelo.
- Diseñan aplicaciones bien fundadas. Al tener en cuenta los atributos de los elementos de datos, podemos reducir la cantidad de descuidos semánticos que podrían generar problemas en el futuro. Los elementos de datos están mejor definidos y las relaciones entre ellos son más completas.
- Reducen los costos y aumentan la eficiencia. Al mitigar el potencial de descuidos, las empresas reducen el riesgo de implementaciones fallidas y la necesidad de revisiones posteriores al lanzamiento. Además, se fomenta la reutilización y el intercambio de datos y se pueden evitar la redundancia y las inconsistencias de datos.
- Proporcionan una base para futuros modelos. Del mismo modo que un modelo de datos conceptual proporciona la base para un modelo de datos lógico, un modelo de datos lógico proporciona el diseño detallado para ser dirigido y ajustado a una tecnología específica en la etapa de modelado de datos físico.

3 Modelado de datos físico

El modelado de datos físico es la tercera de las tres etapas secuenciales del modelado de datos. Los diseñadores de bases de datos producen modelos de datos físicos basándose en los modelos creados en las etapas de modelado de datos conceptual y lógico. Los modelos creados en esta etapa permiten la desnormalización administrada y tienen en cuenta la tecnología de destino para la implementación. Son lo suficientemente minuciosos para representar el diseño de la base de datos tal como se implementó, o como se pretende implementar.

¿Qué es un modelo de datos físico?

Un modelo de datos físico introduce el contexto específico de la base de datos que falta en los modelos de datos lógicos y conceptuales. Representa las tablas, las columnas, los tipos de datos, las vistas, las restricciones, los índices y los procedimientos dentro de la base de datos o la información comunicada durante los procesos informáticos.

Los modelos de datos físicos deben armarse en relación con un sistema de administración de bases de datos (DBMS) específico, así como con los requisitos específicos de los procesos que operan sobre la base de los datos. Esto a menudo requiere la desnormalización de construcciones de diseño lógico para mantener la integridad referencial. Un ejemplo de las consideraciones contextuales en la etapa de modelado de datos físico es la naturaleza de los datos que pueden procesarse/se procesarán y las reglas sobre cómo se pueden ejecutar dichos procesos.

Otra consideración clave es garantizar que los tipos de columna modelados sean compatibles con el DBMS y que se observen las convenciones de nomenclatura para entidades y columnas, lo que evita superposiciones semánticas problemáticas. La consideración del contexto tecnológico significa que los modelos de datos físicos reflejan las necesidades del entorno tecnológico tal como está o según lo previsto.

¿Cuál es el objetivo de un modelo de datos físico?

Como representación específica de la base de datos de la implementación de un modelo de datos, los modelos de datos físicos ayudan a visualizar la estructura de una base de datos antes de que se construya. Su enfoque y objetivo principal es la implementación de una base de datos, y ayudan a las empresas a lograrlo al describir cómo se creará la base de datos dentro de los límites de un DBMS específico.

Con esto, los diseñadores de bases de datos pueden crear una abstracción de la base de datos y generar un esquema. Los tipos de entidades se representan como tablas y las líneas de tipo de relación representan las claves externas entre tablas. Esta perspectiva es fundamental para garantizar que los objetos de datos y las relaciones representadas sean precisas y compatibles con los sistemas de una empresa.

¿Cuándo debería considerar un modelo de datos físico?

Los tres tipos de modelos de datos pueden y deben verse como etapas lineales. Como tercera etapa del modelado de datos, el modelado de datos físico se basa en los modelos desarrollados en las etapas conceptual y lógica.

Los modelos de datos físicos marcan un cambio en los modelos que se construyen principalmente para representar desde "qué" (como en los datos y la información que se modelarán) hasta "cómo" (la implementación). Naturalmente, este enfoque práctico para la implementación tiene en cuenta las características específicas del DBMS y la tecnología, incluidos los requisitos de desnormalización, propuestos para el proyecto.

El modelo describe las necesidades de datos de un solo proyecto, pero también se puede integrar con modelos de datos físicos de otros proyectos para tener en cuenta las interrelaciones entre proyectos, procesos y tecnología. Debido a sus consideraciones de tecnología específica, un modelo de datos físico es más rígido e incluso pequeños cambios pueden requerir modificaciones en toda la aplicación. Por lo tanto, es aconsejable avanzar a la construcción de un modelo de datos físico solo cuando se hayan construido los modelos de datos conceptuales y lógicos.

¿Por qué debería usar un modelo de datos físico?

Los modelos de datos físicos constituyen un paso fundamental para lograr la comprensión de la naturaleza de una implementación. Cuanto más completa sea dicha comprensión, más probabilidades tendrá de implementar con éxito una solución que aborde las necesidades de la empresa.

Un modelo de datos físico bien diseñado da como resultado una mejor calidad de datos, implementaciones y mantenimiento más fáciles y una escalabilidad más alcanzable. Sin embargo, un modelo de datos físico bien diseñado depende de la idoneidad de los modelos que lo preceden. En la práctica, muchas empresas reconocen la necesidad de construir un modelo de datos físico, pero pasan por alto u omiten los modelos conceptuales y lógicos. Esto conduce inevitablemente a lagunas en las consideraciones de diseño y problemas con el linaje y la trazabilidad de los datos desde los modelos de datos hasta las aplicaciones físicas.

Beneficios del modelado de datos físico

Algunos de los beneficios clave de los modelos de datos físicos son los siguientes:

- Ayudan a producir una representación visual de la estructura de la base de datos. Al aprovechar los ejemplos visuales, los diseñadores de bases de datos comprenden mejor los requisitos y pueden comunicar mejor la información de la base de datos a las partes interesadas.
- Reducen el riesgo de implementaciones fallidas o incompletas. Un modelo de datos físico ayuda a una empresa a prepararse mejor y evitar los costos a menudo serios de abordar los descuidos.

- Convierta fácilmente el modelo de datos en un esquema de base de datos. Al utilizar la amplia información y metadatos capturados mientras se construyen los modelos de datos conceptuales, lógicos y físicos, las empresas deben estar preparadas para una traducción 1:1 del modelo de datos en el diseño de la base de datos.

4 Sentencias SQL – DML, DDL, DCL y TCL

¿Qué es DDL? Lenguaje de Definición de Datos

DDL significa Data Definition Language o Lenguaje de Definición de Datos, en español. Este lenguaje permite definir las tareas de las estructuras que almacenarán los datos.

Sentencias de DDL (Data Definition Language)

- CREATE: Utilizado para crear nuevas tablas, campos e índices. Su sintaxis es:
 - CREATE TABLE pasajero
- ALTER: Utilizado para modificar las tablas agregando campos o cambiando la definición de los campos.
 - ALTER TABLE users
- DROP: Empleado para eliminar tablas e índices.
 - DROP TABLE transacciones
- TRUNCATE: Empleado para eliminar todos los registros de una tabla.
 - TRUNCATE TABLE transacciones
- COMMENT: Utilizado para agregar comentarios al diccionario de datos.
- RENAME: Tal como su nombre lo indica es utilizado para renombrar objetos.
 - RENAME TO users_bkup

¿Qué es DML? Lenguaje de Manipulación de Datos

DML significa Data Manipulation Language o Lenguaje de Manipulación de Datos, en español. Este lenguaje permite realizar diferentes acciones a los datos que se encuentran en una base de datos.

Permite recuperar, almacenar, modificar, eliminar, insertar y actualizar datos de una base de datos.

Elementos del DML (Data Manipulation Language)

- **SELECT:** Utilizado para consultar registros de la base de datos que satisfagan un criterio determinado.
 - `SELECT * FROM transacciones`
- **INSERT:** Utilizado para cargar de datos en la base de datos en una única operación.
 - `INSERT INTO trayecto (id_tren, id_estacion, nombre) VALUES (91, 31, 'Konklux');`
- **UPDATE:** Utilizado para modificar los valores de los campos y registros especificados
 - `UPDATE cuentas SET mes='03' WHERE Ciudad='Bogota';`
- **DELETE:** Utilizado para eliminar registros de una tabla de una base de datos.
 - `DELETE FROM transacciones WHERE numero_transacciones=id_cliente;`

¿Qué es DCL? Lenguaje de Control de Datos

Permite crear roles, permisos e integridad referencial, así como el control al acceso a la base de datos.

Elementos del DCL (Data Control Language)

- **GRANT:** Usado para otorgar privilegios de acceso de usuario a la base de datos.
 - `GRANT transacciones TO jose WITH GRANT OPTION`
- **REVOKE:** Utilizado para retirar privilegios de acceso otorgados con el comando GRANT.
 - `REVOKE ALL PRIVILEGES ON transacciones FROM jose;`

¿Qué es TCL? Lenguaje de Control Transaccional

Permite administrar diferentes transacciones que ocurren dentro de una base de datos.

Elementos del TCL**(Transactional Control Language)**

- COMMIT: Empleado para guardar el trabajo hecho.
 - COMMIT transaccion
- ROLLBACK: Utilizado para deshacer la modificación que hice desde el último COMMIT.
 - ROLLBACK transacciones

5 Requerimiento de software

Es la descripción de los servicios y restricciones de un sistema de software, es decir, lo que el software debe hacer y bajo qué circunstancias debe hacerlo.

Tipos de requisitos

- **Requisitos funcionales:** Describen las interacciones entre el sistema y su ambiente, en forma independiente a su implementación. El ambiente incluye al usuario y cualquier otro sistema externo con el cual interactúe el sistema.
- **Requisitos no funcionales:** Describen atributos sólo del sistema o del ambiente del sistema que no están relacionados directamente con los requisitos funcionales. Los requisitos no funcionales incluyen restricciones cuantitativas, como el tiempo de respuesta o precisión, tipo de plataforma (lenguajes de programación y/o sistemas operativos, etc.)

6 Riesgos de software

Si definimos los riesgos de un proyecto como aquellas cuestiones que con cierta probabilidad se podrían convertir en obstáculos para lograr los objetivos, es importante comenzar a trabajar en forma temprana. En algunos libros de administración de proyectos (Steve McConnell, 1996 y Steve McConnell, 1997) se presentan listados de riesgos que pueden ser utilizados como guía.

Sin embargo, no debe olvidarse que son una guía y que posiblemente no concuerde con los riesgos de su proyecto. El trabajo con riesgos es otra de las tareas en la cual es importante contar con cierta madurez para hacer visibles posibles fuentes de riesgo que por diferentes razones no son fáciles de explicitar.

Las fuentes de riesgos en general se deben a:

- **Requerimientos:** la complejidad y la falta de disponibilidad constituyen una fuente de riesgo común en muchos proyectos.

- **Tecnología:** la falta de experiencia en su uso o de madurez de alguna tecnología son fuente de riesgos.
- **Organización desarrolladora:** la falta de apoyo de la gerencia o de compromiso para con el proyecto según fue planificado al momento de competir para ganarlo son fuentes de riesgo frecuentes en los proyectos.
- **Organización cliente:** la falta de experiencia en el involucramiento en proyectos de desarrollo es una fuente frecuente de riesgo.
- **Grupo de desarrollo:** problemas generados por grupos de desarrollo recientemente formados para el proyecto, sin trabajos anteriores compartidos, son fuentes de riesgo comunes en los proyectos.

Los riesgos son importantes porque el líder conducirá la gestión del proyecto a partir de su administración, en el día a día, y además condicionarán la construcción de la estrategia a seguir para dicha gestión (Kurt Bittner, Ian Spence, 2006; Mike Cohn, 2005; McConnell, 1997).

References

- [1] Análisis de requisitos del software. (s/f). Edu.co. Recuperado el 24 de julio de 2022, de <https://tesuva.edu.co/phocadownloadpap/Anlisis%20de%20requisitos%20del%20software.pdf>
- [2] Modelado de datos conceptuales . (s/f). Erwin.com. Recuperado el 24 de julio de 2022, de <https://www.erwin.com/mx-es/solutions/data-modeling/conceptual.aspx>
- [3] Modelado de datos físico. (s/f). Erwin.com. Recuperado el 24 de julio de 2022, de <https://www.erwin.com/mx-es/solutions/data-modeling/physical.aspx>
- [4] Modelado de datos lógico. (s/f). Erwin.com. Recuperado el 24 de julio de 2022, de <https://www.erwin.com/mx-es/solutions/data-modeling/logical.aspx>
- [5] ¿Qué es DDL? Qué significa DML, DCL y TCL + Integridad Referencial. (2017, noviembre 2). Platzi. <https://platzi.com/blog/que-es-ddl-dml-dcl-y-tcl-integridad-referencial/>
- [6] Pantaleo, G. (2011b). Calidad en el desarrollo de software . Alfaomega Grupo Editor Argentino, 2011.