

## **1. ¿Cuál es la última versión de SQL?**

ISO/IEC 9075:2016 o SQL:2016 es un estándar dedicado al ámbito de las tecnologías de la información, y más concretamente al lenguaje para gestión de bases de datos SQL. Fue publicado en diciembre de 2016, sustituyendo a la revisión de 2011, SQL:2011, convirtiéndose así en la referencia actual del lenguaje de consultas estructuradas (SQL).

SQL: 2016 consta de nueve partes [1] - [9], todas las cuales fueron publicadas juntas. Sin embargo, con la excepción de la Parte 2, "Fundación", las otras partes no cambian significativamente de sus versiones anteriores, que contiene principalmente correcciones de errores y cambios necesarios para alinearse con la nueva funcionalidad en la Parte 2. Al igual que con las versiones anteriores, SQL: 2016 está disponible para su compra en las tiendas web ANSI2 e ISO3.

Las principales características nuevas de SQL: 2016 son:

- Soporte para la notación de objetos de Java Script (JSON) datos
- Funciones de tabla polimórfica
- Reconocimiento de patrones de filas

SQL: 2016 también incluye una serie de funciones más pequeñas, como funciones adicionales integradas. Además del estándar SQL formal, el SQL comité ha desarrollado una serie de informes técnicos (TR). Los TR, aunque no son normativos, contienen información que es útil para comprender cómo el estándar SQL trabaja.

## **2. Primer modelo de datos**

### **Modelo jerárquico de base de datos**

Este es el modelo más antiguo, en las bases de datos jerárquicas las dependencias son inequívocas. Cada registro tiene solo un precedente (Parent-Child Relationships, PCR) a excepción de la raíz (root), constituyendo un esquema en árbol como el de arriba. Mientras que cada nodo "hijo", solo puede tener un nodo "padre", los "padres" pueden tener tantos "hijos" como quieran. Dado el estricto ordenamiento jerárquico, los niveles sin relación directa, no interactúan entre sí y conectar dos árboles diferentes tampoco es fácil. Por todo esto, las estructuras de base de datos jerárquicas son extremadamente inflexibles, pero muy claras.

Los registros con hijos se llaman records y los que no tienen se llaman hojas y son los que suelen contener los documentos. Los records sirven para clasificar las hojas. Las consultas a una base de datos jerárquica alcanzan a las hojas partiendo desde la raíz y pasando por los distintos records.

### 3. Organización o institución que aprueba todo sobre BD

Un **administrador de bases de datos** también conocido como **DBA** (*database administrator*) es aquel profesional que administra las tecnologías de la información y la comunicación, siendo responsable de los aspectos técnicos, tecnológicos, científicos, inteligencia de negocios y legales de bases de datos, y de la calidad de datos.

Sus tareas incluyen las siguientes:

- Implementar, dar soporte y gestionar bases de datos corporativas.
- Crear y configurar bases de datos relacionales.
- Ser responsables de la integridad de los datos y la disponibilidad.
- Diseñar, desplegar y monitorizar servidores de bases de datos.
- Diseñar la distribución de los datos y las soluciones de almacenamiento.
- Garantizar la seguridad de las bases de datos, realizar copias de seguridad y llevar a cabo la recuperación de desastres.
- Planificar e implementar el aprovisionamiento de los datos y aplicaciones.
- Diseñar planes de contingencia.
- Diseñar y crear las bases de datos corporativas de soluciones avanzadas.
- Analizar y reportar datos corporativos que ayuden a la toma de decisiones en la inteligencia de negocios.
- Producir diagramas de entidades relacionales y diagramas de flujos de datos, normalización esquemática, localización lógica y física de bases de datos y parámetros de tablas.

### 4. ¿Qué hay en el área de bases de datos actualmente?

#### Base de datos distribuida

La base de datos distribuida consiste en el almacenamiento de porciones de la base de datos en diferentes ubicaciones físicas y, por tanto, el procesamiento está distribuido o replicado entre los distintos puntos de una red de trabajo.

Consecuentemente, la base de datos distribuida tiene una mayor disponibilidad de los datos debido a sus múltiples ubicaciones. De esta forma, si una de las bases de datos falla, podría seguir funcionando correctamente, aunque puede que un poco ralentizado. Sin embargo, esta misma característica de estar distribuido puede provocar que haya duplicidad de los datos y un menor nivel de seguridad.

Dentro de las bases de datos distribuidas hay 2 tipos: homogéneas y heterogéneas. Las homogéneas son conscientes de las otras ubicaciones y cooperan en el procesamiento de las solicitudes. Además, tienen el mismo esquema y el mismo sistema de gestión de base de datos (DBMS). Mientras que las heterogéneas

cooperan de forma limitada el procesamiento ya que pueden tener el DBMS y los esquemas diferentes e, incluso, que los sitios no se conozcan entre sí.

### **Base de datos NoSQL**

El nombre de la base de datos NoSQL proviene de Not only SQL o, en español, no solo SQL. Esto se debe a que este tipo de base de datos suele evitar el uso del SQL o lo usa de apoyo, pero no como consulta. El hecho de evitar el SQL es porque se usa para proyectos en los que se necesita trabajar en la base de datos con un gran volumen. En las bases de datos con lenguaje SQL, los distintos atributos de un elemento, están en diferentes columnas, mientras que en una NoSQL todos los atributos se encuentran en una misma columna, ahorrando espacio. Algunos ejemplos de lenguajes usados por bases de datos NoSQL son: JSON (JavaScript Object Notation); CQL (Contextual Query Language, anteriormente conocido como Common Query Language); o GQL (Graph Query Language). Además, este tipo de bases de datos no suele permitir las uniones lógicas o joins. Esta restricción se debe al extremadamente alto volumen de datos que suelen manejar.

### **Base de datos orientada a objetos**

La base de datos orientada a objetos, o Object-Oriented Database, representa los datos en forma de objetos y clases. El objeto puede ser desde un resultado de búsqueda a una tabla; y una clase es una colección de objetos.

Los objetos tienen la capacidad de encapsular tanto un estado como un comportamiento de un objeto. Además, también puede almacenar las relaciones que tiene con otros objetos e, incluso, agruparse con otros objetos para formar objetos complejos. De esta forma, el objeto puede ser referenciado o nombrado posteriormente, como una unidad sin tener que entrar en sus complejidades. Este tipo de base de datos están generalmente escritas en lenguajes de programación orientados a objetos como Java, C o Smalltalk.

### **Base de datos gráfica o de grafos**

En cuanto a la base de datos gráfica, también conocida como orientada a grafos, se diferencian de las anteriores en que están especializadas en establecer relaciones entre los datos de forma visual y navegar por dichas relaciones. Para leer la información, hay que leer los nodos o conectores (puntos de conexión de los datos de las tablas), generando un lenguaje natural. Algunos ejemplos de bases de datos de gráficas son Neo4j y Amazon Neptune.

### **Bases de datos jerárquicas**

En este modelo los datos se organizan en una forma similar a un árbol, en donde un nodo padre de información puede tener varios hijos. El nodo que no tiene padres es llamado raíz, y a los nodos que no tienen hijos se los conoce como hojas. Las bases de datos jerárquicas son especialmente útiles en el caso de aplicaciones

que manejan un gran volumen de información y datos muy compartidos permitiendo crear estructuras estables y de gran rendimiento.

Una de las principales limitaciones de este modelo es su incapacidad de representar eficientemente la redundancia de datos.

### **Base de datos de red**

Éste es un modelo ligeramente distinto del jerárquico; su diferencia fundamental es la modificación del concepto de nodo: se permite que un mismo nodo tenga varios padres (posibilidad no permitida en el modelo jerárquico).

Fue una gran mejora con respecto al modelo jerárquico, ya que ofrecía una solución eficiente al problema de redundancia de datos; pero, aun así, la dificultad que significa administrar la información en una base de datos de red ha significado que sea un modelo utilizado en su mayoría por programadores más que por usuarios finales.

### **Bases de datos transaccionales**

Son bases de datos cuyo único fin es el envío y recepción de datos a grandes velocidades, estas bases son muy poco comunes y están dirigidas por lo general al entorno de análisis de calidad, datos de producción e industrial, es importante entender que su fin único es recolectar y recuperar los datos a la mayor velocidad posible, por lo tanto la redundancia y duplicación de información no es un problema como con las demás bases de datos, por lo general para poderlas aprovechar al máximo permiten algún tipo de conectividad a bases de datos relacionales.

### **Bases de datos relacionales**

Éste es el modelo utilizado en la actualidad para modelar problemas reales y administrar datos dinámicamente. Tras ser postulados sus fundamentos en 1970 por Edgar Frank Codd, de los laboratorios IBM en San José (California), no tardó en consolidarse como un nuevo paradigma en los modelos de base de datos. En este modelo, el lugar y la forma en que se almacenen los datos no tienen relevancia. Esto tiene la considerable ventaja de que es más fácil de entender y de utilizar para un usuario esporádico de la base de datos. La información puede ser recuperada o almacenada mediante "consultas" que ofrecen una amplia flexibilidad y poder para administrar la información.

El lenguaje más habitual para construir las consultas a bases de datos relacionales es SQL, Structured Query Language o Lenguaje Estructurado de Consultas, un estándar implementado por los principales motores o sistemas de gestión de bases de datos relacionales.

## Referencias

<https://www.researchgate.net/publication/329593276> The new and improved SQL2016 standard

<https://www.ticportal.es/glosario-tic/base-datos-database>

<http://www3.uji.es/~mmarques/proyecto/node4.html>

<https://histinf.blogs.upv.es/2011/01/04/historia-de-las-bases-de-datos/>

<https://www.ionos.mx/digitalguide/hosting/cuestiones-tecnicas/bases-de-datos/>