



ADMINISTRANDO BASES DE DATOS CON SQL SERVER 2016

Descripción breve

Los asistentes a este webinar identificarán los diferentes componentes que presenta SQL Server de Microsoft para la correcta administración de las bases de datos contenidas en este gestor

Isaias Islas Gonzalez
Isaias.islas@live.com.mx 55 5402 0974



Administrando bases de datos con SQL Server 2016



Expositor: [Ing. Isaías Islas González \(CDMX\)](#)

Fecha: febrero 2022

- [¿Qué es la ADMINISTRACION?](#)
- [¿Qué es una base de datos?](#)
 - [Primera base de datos](#)
- [¿Qué hace un DBA en IT?](#)
- [¿Qué es el lenguaje de consulta estructurada SQL?](#)
 - [DML, DDL, DCL, TCL](#)
 - [ANSI](#)
- [Evolución de las bases de datos](#)
- [Tipos de bases de datos](#)
 - [Stretch Database](#)
- [¿Qué es un software de base de datos?](#)
 - [SQL Server Management Studio \(SSME\)](#)
- [¿Qué es SQL Server de Microsoft?](#)
- [Componentes de SQL Server](#)
- [¿Datos vs Información?](#)
- Modelando bases de datos con ERWIN MODEL MART (taller) – [Escenario de Uso](#)
- [Un poco de historia de SQL Server](#)
- [Ciclo de vida del producto](#)
- [Evolución de SQL Server](#) en las diferentes versiones
- [Especificaciones máximas de SQL Server](#)
- [Instalación de SQL Server 2019](#) (Paso a paso)
 - [¿Cuántas ediciones tiene SQL Server 2019?](#)
 - [SQL Server Management Studio \(SSME\)](#)
 - [Requisitos para la instalación de SQL Server 2016](#)
 - [Libros en línea \(BOL\)](#)
 - [Instalación desatendida](#)
 - [¿Qué CARACTERISTICAS \(componentes\) de SQL Server debo instalar?](#)
 - [Mejores prácticas para la configuración de la base TEMPDB](#)
 - [SQL Server Best Practice Analyzer Download](#)
 - [Base Line Configuration Analyzer](#)
 - [Detener o arrancar los servicios de SQL Server – Por comando](#)
 - [SQL Server Configuration Manager - revisando o cambiando la configuración de mi SQL Server](#)
 - [Abrir el SQL Server Configuration Manager Personal](#)



- [¿Y que es el servicio de AGENT?](#)
- [¿Qué son las bases de datos del sistema?](#)
 - [Mover mi base MASTER a otro disco](#)
 - [Mover mis bases de datos del sistema a otro disco](#)
- [¿Cómo se comportan los respaldos y cuantos tipos hay en SQL Server?](#)
 - [Descripción de los tipos de copia de seguridad de SQL Server](#)
 - [5 métodos de respaldo en SQL Server 2019](#)
 - [Copia de seguridad del log en un ambiente Availability Groups \(AG\)](#)
 - [Realizar un plan de respaldos mediante un plan de mantenimiento](#)
- [Una vista dentro de la cache del buffer](#)
 - [Buffer Cache Hit ratio y Page life expectancy](#)
- ¿Qué hacer después de una Restauración de una base de datos?
 - [¿Qué es un INDICE?](#)



- Regeneración de índices
 - [¿Qué es un índice? Video: Victor Hugo Cardenas](#)
 - [Uso de los índices \(video\)](#)
 - [¿Cuántos tipos de índices maneja SQL Server?](#)
 - [Hacer uso de Database Tuning Advisor \(DTA\) para crear índices y estadísticas](#)
- Regeneración de estadísticos
 - [¿Qué son las estadísticas?](#)
 - [¿Qué son las LIVE QUERY STATISTICS?](#)
- Tránsito de Usuarios (Paso a paso)
 - [Contained Databases \(Independientes\)](#)
 - [SSIS Transfer Logic Task](#) (Transfiriendo usuarios o Login entre instancias)
 - [Asistente copia de Base de Datos](#)
 - [Script de Microsoft sp_hexadecimal](#)
- [Creación de Planes de mantenimiento \(manual\)](#)
 - [Mediante el SSME \(wizard\)](#)
 - [SQL Server Data Tools Download](#)
- Creación de una Traza
 - [Database Experimentation Assistant \(DEA\) Download](#)
 - [Como utilizar la herramienta DEA](#)
- Migrando bases de datos de una versión inferior a una versión superior
 - [Platicando sobre el NIVEL DE COMPATIBILIDAD](#)
 - Utilizando una forma tradicional o profesional



- [Utilizando Database Migration Assistant \(DMA\)](#)
- Objetos de una base de datos
 - [Relación de objetos de una base de datos en SQL Server](#)
 - [Tablas temporales versionadas](#)
 - [Dependencias de un objeto en SQL Server](#)
- Ejecución de consultas para obtener:
 - Las consultas más pesadas
 - Las consultas más usadas
 - Las tablas sin índices
 - Las tablas sin Primary Key
 - Las tablas sin Foreign Key
 - Las tablas sin un IDENTITY
 - Los bloqueos entre consultas o transacciones
 - [Planes de ejecución, índices y estadísticas \(video 1:56\)](#)
 - [SQL Sentry Plan Explorer Download](#)
- [Tuning advisor \(Ejemplo\)](#)
- [Particionamiento de tablas \(Video 16 minutos\)](#)
 - [Ejemplo de particionamiento de tablas](#)
 - [Conceptos básicos sobre particionamiento](#)
- [Bases de tipo instantáneas \(Snapshots\)](#)
 - Información general
 - Ventajas
 - Términos y definiciones
 - Requisitos previos
 - Limitaciones de las bases de origen
 - Limitaciones de las instantáneas
 - Requisitos de espacio en disco
 - Instantáneas de base de datos con grupos de archivos sin conexión
 - [Ejemplo de creación de una base de tipo snapshot](#)
- [Roles de servidor y bases de datos](#)
 - [¿Puedo crear un rol a nivel servidor?](#)
 - [¿Puedo crear un rol a nivel base de datos?](#)
- [Alta Disponibilidad](#)
 - ¿DRP o HA?, Diferencias
 - Backup – Restore
 - [Creando un Failover Cluster Instance \(FCI\) Virtualizado \(paso a paso\)](#)
 - [Configurando un esquema de Mirroring \(Step by Step\)](#)
 - [AlwaysOn \(Victor Hugo Cardenas – 51 minutos - Video\)](#)
 - [¿Replicación?](#)
 - [Replicación step by step](#)
- ¿Nos vas a mencionar algo sobre ALERTAS?
 - [Como configurar el sistema de correos en SQL Server](#)
 - [Las ALERTAS default \(14 en total mediante T-SQL\)](#)
 - [Alerta de espacio en disco y en bases de datos \(T-SQL\)](#)
 - [¿Cómo envió el resultado de una consulta \(query\)?](#)



- ¿Existe **AUDITORIA** en SQL Server?
 - [Creación de una especificación de auditoria para servidor y base de datos](#)
 - [Auditorias en SQL Server – Paso a Paso \(Maxi de Accotto\)](#)
- ¿Qué es **DQS** en SQL server?
 - [Introducción a Data Quality Services](#)
 - [Construyendo un Data Quality Knowledge Base con SSDQS Studio](#)
 - [Reglas de coincidencia \(matching\) y registros duplicados utilizando DQS](#)
 - [Automatizando los procesos de Matching en DQS](#)
- [¿Qué son los EVENTOS EXTENDIDOS \(xEvents\)?](#)
- [¿Hablamos de CDC \(Change Data Capture\)?](#)
 - [¿Se puede automatizar?](#)
- [Instalando SQL Server Integration Services paso a paso](#)
- [¿Qué cambios o mejoras hubo en 2016?](#)
- [¿Qué hay de nuevo en SQL Server 2019 \(Victor Hugo Cardenas\)](#)
 - [10 Mejoras en SQL Server 2019 para desarrolladores](#)





¿Qué es una base de datos?

Una base de datos es una recopilación organizada de información o datos estructurados, que normalmente se almacena de forma electrónica en un sistema informático. Normalmente, una base de datos está controlada por un [sistema de gestión de bases de datos \(DBMS\)](#). En conjunto, los datos y el DBMS, junto con las aplicaciones asociadas a ellos, reciben el nombre de sistema de bases de datos, abreviado normalmente a simplemente base de datos.

Los datos de los tipos más comunes de bases de datos en funcionamiento actualmente se suelen utilizar como estructuras de filas y columnas en una serie de tablas para aumentar la eficacia del procesamiento y la consulta de datos. Así, se puede acceder, gestionar, modificar, actualizar, controlar y organizar fácilmente los datos. La mayoría de las bases de datos utilizan un lenguaje de consulta estructurada (SQL) para escribir y consultar datos. [¿Cuál fue la primera base de datos del ser humano?](#)

¿Qué es el lenguaje de consulta estructurada (SQL)?

El SQL es un lenguaje de programación que utilizan casi todas las [bases de datos relacionales](#) para consultar, manipular y definir los datos, además de proporcionar control de acceso. El SQL se desarrolló por primera vez en IBM en la década de 1970 con Oracle como uno de los principales contribuyentes, lo que dio lugar a la implementación del estándar ANSI SQL. El SQL ha propiciado muchas ampliaciones de empresas como IBM, Oracle y Microsoft. Aunque el SQL se sigue utilizando mucho hoy en día, están empezando a aparecer nuevos lenguajes de programación.

Evolución de la base de datos

Las bases de datos han evolucionado drásticamente desde su inicio a principios de la década de 1960. Las bases de datos de navegación, como la base de datos jerárquica (que se basaba en un modelo de árbol y permitía una relación de uno a muchos) y la base de datos de red (un modelo más flexible que permitía relaciones múltiples), eran los sistemas originales que se utilizaban para almacenar y manipular datos. Aunque eran sencillos, estos primeros sistemas eran inflexibles. En la década de 1980, se hicieron populares las [bases de datos relacionales](#), seguidas de las [bases de datos orientadas a objetos](#) en la década de 1990. Más recientemente, las [bases de datos NoSQL](#) surgieron como respuesta al crecimiento de Internet y la necesidad de acelerar la velocidad y el procesamiento de los datos no estructurados. Hoy en día, las [bases de datos en la nube](#) y las [bases de datos de autogestión](#) están abriendo



nuevos horizontes en lo que respecta a la forma en la que se recopilan, se almacenan, se gestionan y se utilizan los datos.

Tipos de bases de datos

Existen muchos tipos diferentes de bases de datos. La mejor base de datos para una organización específica depende de cómo pretenda la organización utilizar los datos.

- **Bases de datos relacionales.** Las bases de datos se hicieron predominantes en la década de 1980. Los elementos de una base de datos relacional se organizan como un conjunto de tablas con columnas y filas. La tecnología de bases de datos relacionales proporciona la forma más eficiente y flexible de acceder a información estructurada.
- **Bases de datos orientadas a objetos.** La información de una base de datos orientada a objetos se representa en forma de objetos, como en la programación orientada a objetos.
- **Bases de datos distribuidas.** Una base de datos distribuida consta de dos o más archivos que se encuentran en sitios diferentes. La base de datos puede almacenarse en varios ordenadores, ubicarse en la misma ubicación física o repartirse en diferentes redes.
- **Almacenes de datos.** Un repositorio central de datos, un data warehouse es un tipo de base de datos diseñado específicamente para consultas y análisis rápidos.
- **Bases de datos NoSQL.** Una base de datos [NoSQL](#), o base de datos no relacional, permite almacenar y manipular datos no estructurados y semiestructurados (a diferencia de una base de datos relacional, que define cómo se deben componer todos los datos insertados en la base de datos). Las bases de datos NoSQL se hicieron populares a medida que las aplicaciones web se volvían más comunes y complejas.
- **Bases de datos orientadas a grafos.** Una base de datos orientada a grafos almacena datos relacionados con entidades y las relaciones entre entidades.
- **Bases de datos OLTP.** Una base de datos OLTP es una base de datos rápida y analítica diseñada para que muchos usuarios realicen un gran número de transacciones.

Estos son solo algunos de las varias docenas de tipos de bases de datos que se utilizan hoy en día. Otras bases de datos menos comunes se adaptan a funciones científicas, financieras o de otro tipo muy específicas. Además de los diferentes tipos de bases de datos, los cambios en los enfoques de desarrollo tecnológico y los avances considerables, como la nube y la automatización, están impulsando a las bases de datos en direcciones completamente nuevas. Algunas de las bases de datos más recientes incluyen

- **Bases de datos de código abierto.** Un sistema de base de datos de código abierto es aquel cuyo código fuente es de código abierto; tales bases de datos pueden ser bases de datos SQL o NoSQL.



- **Bases de datos en la nube.** Una [base de datos en la nube](#) es una recopilación de datos, estructurados o no estructurados, que reside en una plataforma de cloud computing privada, pública o híbrida. Existen dos tipos de modelos de bases de datos en la nube: tradicional y base de datos como servicio (DBaaS). Con DBaaS, un proveedor de servicios realiza las tareas administrativas y el mantenimiento.
- **Base de datos multimodelo.** Las bases de datos multimodelo combinan distintos tipos de modelos de bases de datos en un único servidor integrado. Esto significa que pueden incorporar diferentes tipos de datos.
- **Bases de datos de documentos/JSON.** Diseñadas para almacenar, recuperar y gestionar información orientada a los documentos, las bases de datos de documentos son una forma moderna de almacenar los datos en formato JSON en lugar de en filas y columnas.
- **Bases de datos de autogestión.** El tipo de base de datos más nuevo e innovador, las bases de datos de autogestión (también conocidas como bases de datos autónomas) están basadas en la nube y utilizan el machine learning para automatizar el ajuste de la base de datos, la seguridad, las copias de seguridad, las actualizaciones y otras tareas de gestión rutinarias que tradicionalmente realizan los administradores de bases de datos.

¿Qué es un software de base de datos?

El software de base de datos se utiliza para crear, editar y mantener archivos y registros de bases de datos, lo que facilita la creación de archivos y registros, la entrada de datos, la edición de datos, la actualización y la creación de informes. El software también maneja el almacenamiento de datos, las copias de seguridad y la creación de informes, así como el control de acceso múltiple y la seguridad. La seguridad sólida de las bases de datos es especialmente importante hoy en día, ya que el robo de datos es cada vez más frecuente. En ocasiones, el software de base de datos también se denomina "sistema de gestión de bases de datos" ([DBMS](#)).

El software de base de datos simplifica la gestión de datos, ya que permite a los usuarios almacenar datos de forma estructurada y acceder posteriormente a ellos. Por lo general, tiene una interfaz gráfica que ayuda a crear y administrar los datos y, en algunos casos, los usuarios pueden [construir sus propias bases de datos](#) mediante el uso de software de base de datos.

¿Qué es un sistema de gestión de bases de datos ([DBMS](#))?

Normalmente, una base de datos requiere un programa de software de bases de datos completo, conocido como sistema de gestión de bases de datos (DBMS). Un DBMS sirve como interfaz entre la base de datos y sus programas o usuarios finales,



lo que permite a los usuarios recuperar, actualizar y gestionar cómo se organiza y se optimiza la información. Un DBMS también facilita la supervisión y el control de las bases de datos, lo que permite una variedad de operaciones administrativas como la supervisión del rendimiento, el ajuste, la copia de seguridad y la recuperación.

Algunos ejemplos de software de bases de datos o DBMS populares incluyen MySQL, Microsoft Access, Microsoft SQL Server, FileMaker Pro, Oracle Database y dBASE.

¿Qué es SQL Server de Microsoft?

Hoy en día es prácticamente impensable desarrollar una aplicación sin que esta trabaje en conjunto con una base de datos. Para entender qué es una base de datos, podemos definirlo como un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para un posterior uso. Por ejemplo, una biblioteca puede considerarse una base de datos compuesta en su mayoría por documentos y textos impresos en papel e indexados para su consulta.

En el mundo empresarial existen programas denominados "sistemas gestores de bases de datos", abreviado SGBD (del inglés Database Management System o DBMS), que permiten almacenar y posteriormente acceder a los datos de forma rápida y estructurada. En el mercado existen varias alternativas, algunas libres, como las ofrecidas por marcas como Microsoft, Oracle, Sybase ASE, PostgreSQL o MySQL. Microsoft SQL Server es la alternativa de Microsoft a otros potentes sistemas gestores de bases de datos. Es un sistema de gestión de base de datos relacional desarrollado como un servidor que da servicio a otras aplicaciones de software que pueden funcionar ya sea en el mismo ordenador o en otro ordenador a través de una red (incluyendo Internet).

Breve historia de SQL Server Microsoft

La historia de SQL (que se pronuncia deletreando en [inglés](#) las letras que lo componen, es decir "ese-cu-ele" y no "siquel" como se oye a menudo) empieza en 1974 con la definición, por parte de **Donald Chamberlin** y de otras personas que trabajaban en los laboratorios de [investigación](#) de IBM, de un lenguaje para la especificación de las características de las bases de datos que adoptaban el [modelo](#) relacional. Este lenguaje se llamaba SEQUEL (Structured English Query Language) y se implementó en un prototipo llamado SEQUEL-XRM entre 1974 y 1975. Las experimentaciones con ese prototipo condujeron, entre 1976 y 1977, a una revisión del lenguaje (SEQUEL/2), que a partir de ese momento cambió de nombre



por motivos legales, convirtiéndose en SQL. El prototipo (System R), basado en este lenguaje, se adoptó y utilizó internamente en IBM y lo adoptaron algunos de sus [clientes](#) elegidos. Gracias al [éxito](#) de este [sistema](#), que no estaba todavía comercializado, también otras compañías empezaron a desarrollar sus [productos](#) relacionales basados en SQL. A partir de 1981, IBM comenzó a entregar sus productos relacionales y en 1983 empezó a vender DB2. En el curso de los años ochenta, numerosas compañías (por ejemplo Oracle y Sybase, sólo por citar algunos) comercializaron productos basados en SQL, que se convierte en el estándar industrial de hecho por lo que respecta a las bases de datos relacionales.

En 1986, el ANSI adoptó SQL (sustancialmente adoptó el dialecto SQL de IBM) como estándar para los lenguajes relacionales y en 1987 se transformó en estándar [ISO](#). Esta versión del estándar va con el nombre de SQL/86. En los años siguientes, éste ha sufrido diversas revisiones que han conducido primero a la versión SQL/89 y, posteriormente, a la actual SQL/92. El hecho de tener un estándar definido por un lenguaje para bases de datos relacionales abre potencialmente el camino a la Inter comunicabilidad entre todos los productos que se basan en él. Desde el punto de vista práctico, por desgracia las cosas fueron de otro modo. Efectivamente, en general cada productor adopta e implementa en la propia base de datos sólo el [corazón](#) del lenguaje SQL (el así llamado Entry level o al máximo el Intermediate level), extendiéndolo de manera individual según la propia visión que cada cual tenga del mundo de las bases de datos. Actualmente, está en marcha un [proceso](#) de revisión del lenguaje por parte de los comités ANSI e ISO, que debería terminar en la definición de lo que en este momento se conoce como SQL3. Las características principales de esta nueva encarnación de SQL deberían ser su transformación en un lenguaje stand-alone (mientras ahora se usa como lenguaje hospedado en otros lenguajes) y la [introducción](#) de nuevos [tipos de datos](#) más complejos que permitan, por ejemplo, el tratamiento de datos multimediales.

Administración del ciclo de vida de base de datos

Administración de ciclo de vida de base de datos (DLM) es un enfoque basado en directiva para administrar bases de datos y activos de datos. DLM no es un producto sino un enfoque integral para administrar el esquema de la base de datos, los datos y los metadatos para una aplicación de base de datos. Un enfoque proactivo y reflexivo de DLM permite a una organización administrar los recursos de datos según los niveles adecuados de rendimiento, protección, disponibilidad y costo.

DLM comienza con la descripción del diseño y el propósito del proyecto, continúa con las actividades de desarrollo, prueba, compilación, implementación,



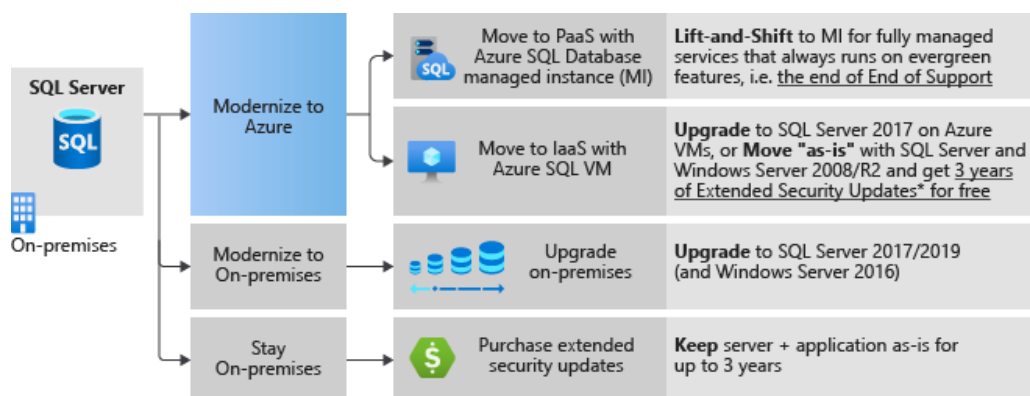
mantenimiento, supervisión y copia de seguridad de la base de datos, y finaliza con el archivado de datos. En este tema se proporciona información general de las fases de DLM que comienzan con el desarrollo de la base de datos y continúan con las acciones de compilación, implementación y supervisión. También se incluyen actividades de administración de datos y operaciones de portabilidad de datos como importación/exportación, copia de seguridad, migración y sincronización.

Evolución SQL Server Microsoft

La siguiente imagen proporciona un ejemplo de innovación y evolución en las distintas versiones de SQL Server a lo largo de los años:

	Performance and productivity	Self-service BI	Cloud-ready	Mission critical and cloud performance	Built-in ML	Docker and Linux	Spark & HDFS
SQL Server 2008	Compression	Policy-based mgnt.	Programmability				
SQL Server 2008 R2	PowerPivot (In-Memory)	SharePoint integration	Master Data Services				
SQL Server 2012	AlwaysOn	In-Memory ColumnStore	Data Quality Services	Power View	Cloud		
SQL Server 2014	In-memory across workloads	Hybrid cloud optimized	HDInsight service	Cloud BI			
SQL Server 2016	Industry leading performance	Built-in ML	Hybrid cloud optimized	Cloud Engineering First	Cloud BI		
SQL Server 2017	Industry leading performance	Graph DB	Built-in ML	Choice of language & platform	Easy migration to the cloud		
SQL Server 2019	Industry leading performance	ML-enabled perf & security	Built-in ML	Choice of language & platform	Lift & shift to the cloud		

Para obtener más información, orientación y herramientas para planificar y automatizar su actualización o migración, consulte [Fin de soporte de SQL Server 2005](#) y [Fin de soporte de SQL Server 2008](#).





QUIÉN SOY



ISAIAS ISLAS GONZALEZ

Cuenta HSBC: 6323256969

Clabe: 021180063232569695

ISAIAS ISLAS GONZALEZ

- Master Consultant SQL Server 4.0 a 2019
- ESIME (IPN) / 83
- MCP MCTS MPN ADW
- Especialidad en HA
- Empresas
 - Grupo Carso 80 – 90
 - Super Soya 91-93
 - Grupo Salinas 95-06
 - Banca Mifel 06-10
 - Datavision (EMAIL: isaias.islas@datavision.com.mx)
 - Correo personal: isaias.islas@live.com.mx