

# Jesus Gil

## MVP Data Platform



@jesus\_gilv



[linkedin.com/in/jesúsgilv/](https://linkedin.com/in/jesúsgilv/)



[github.com/jesusgilv/](https://github.com/jesusgilv/)

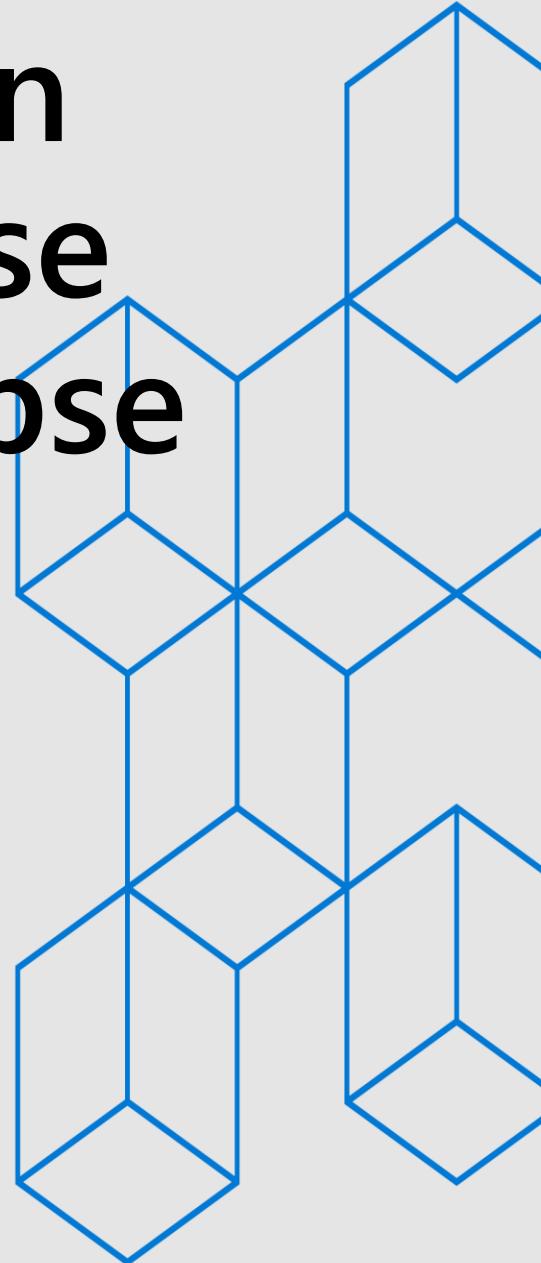


SQLRudo



# Implementa un Data Warehouse con Azure Synapse Analytics

Jesus Gil | Dr Rudo SQL  
@jesus\_gilv | @SQL\_Rudo





# ¿Qué es Azure Synapse Analytics (anteriormente SQL Data Warehouse)?

- Azure Synapse es un servicio de análisis que engloba el almacenamiento de datos empresariales y el análisis de macrodatos.
- Ofrece la libertad de consultar los datos como prefiera, ya sea a petición sin servidor o con recursos aprovisionados, a escala.
- Azure Synapse reúne estos dos mundos con una experiencia unificada para ingerir, preparar, administrar y servir datos para las necesidades inmediatas de inteligencia empresarial y aprendizaje automático.

# Pasos típicos de una implementación



Diseño de un  
almacenamiento de datos  
con Azure Synapse Analytics



Consulta de datos en Azure  
Synapse Analytics



Importación de datos en  
Azure Synapse Analytics  
mediante PolyBase

# Azure Synapse tiene 4 componentes

- SQL de Synapse: Análisis basado en T-SQL completo: disponible con carácter general
  - Grupo de SQL (pago por DWU aprovisionado)
  - SQL a petición (pago por TB procesados): (versión preliminar).
- Spark: profunda integración con Apache Spark (versión preliminar).
- Canalizaciones de Synapse: integración de datos híbridos (versión preliminar).
- Studio: Experiencia de usuario unificada (versión preliminar)



# Grupo de SQL de Synapse en Azure Synapse

- El grupo de SQL de Synapse hace referencia a las características de almacenamiento de datos empresariales que están disponibles con carácter general en Azure Synapse.
- El grupo de SQL representa una colección de recursos de análisis que se aprovisionan al usar SQL de Synapse. El tamaño del grupo de SQL lo determinan las unidades de almacenamiento de datos (DWU).
- Importe macrodatos con consultas T-SQL de [PolyBase](#) simples y, después, use la potencia de MPP para realizar análisis de alto rendimiento. Al realizar la integración y el análisis, el grupo de SQL de Synapse pasará a ser la versión única de certeza con la que puede contar su empresa para obtener información.

# Componente clave de una solución de macrodatos

El almacenamiento de datos un componente clave de una solución de macrodatos de un extremo a otro en la nube.



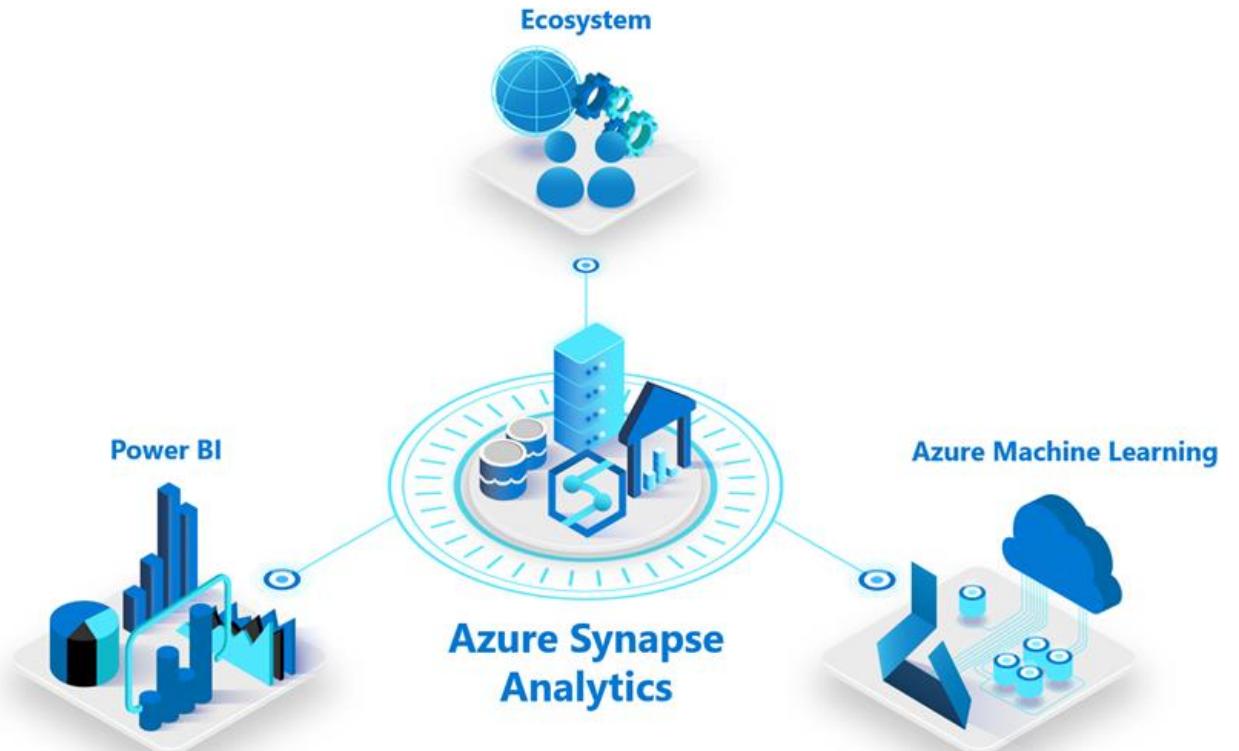


# Componente clave de una solución de macrodatos

- En una solución de datos en la nube, los datos provenientes de varios orígenes se ingieren en almacenes de macrodatos. Una vez que están en un almacén de macrodatos, Hadoop, y Spark y los algoritmos de aprendizaje automático preparan y entrena los datos. Cuando los datos están listos para el análisis complejo, el grupo de SQL de Synapse usa PolyBase para consultar los almacenes de macrodatos. PolyBase usa las consultas T-SQL estándar para llevar los datos a tablas del grupo de SQL de Synapse.
- El grupo de SQL de Synapse almacena los datos en tablas relacionales con almacenamiento en columnas. Este formato reduce considerablemente los costos de almacenamiento de datos y mejora el rendimiento de las consultas. Una vez que los datos están almacenados, se pueden realizar análisis a gran escala. En comparación con los sistemas de bases de datos tradicionales, las consultas de análisis finalizan en segundos, en lugar de minutos, o en horas, en lugar de días.
- Los resultados del análisis pueden ir a aplicaciones o bases de datos de informes mundiales. En ellas, los analistas empresariales pueden obtener información útil para tomar decisiones empresariales bien informadas.

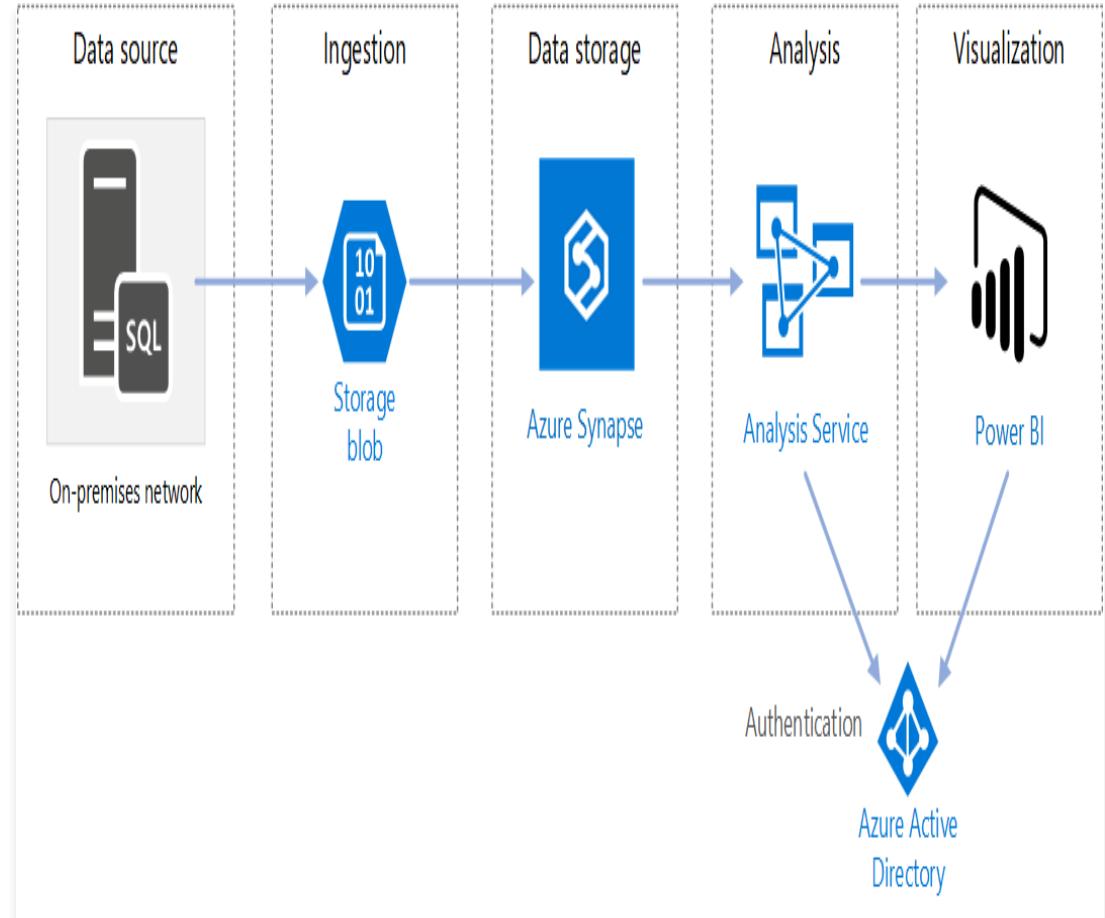
# Arquitectura de Azure Synapse Analytics

- SQL Analytics
- Spark
- Integración de datos
- Studio



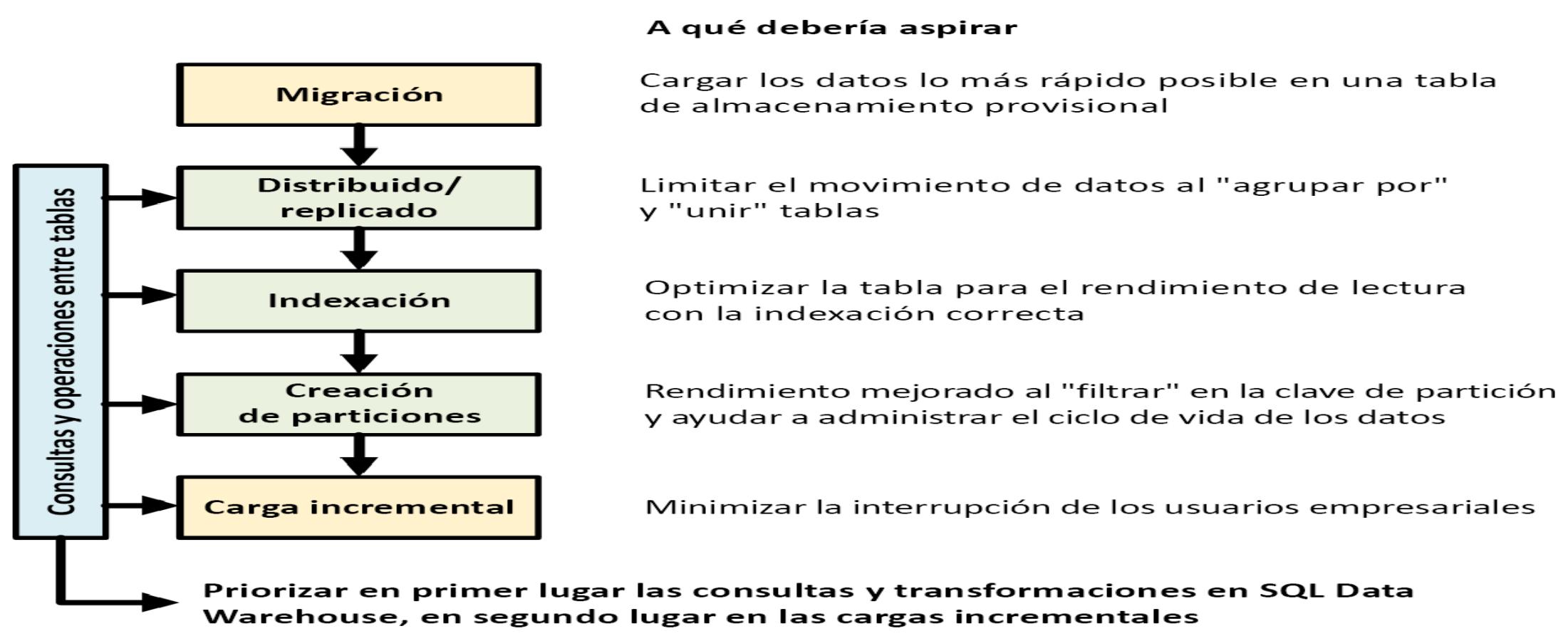
# Enterprise BI en Azure con Azure Synapse Analytics

- Esta arquitectura de referencia implementa una canalización de extracción, carga y transformación (ELT) que mueve los datos de una base de datos de SQL Server local a Azure Synapse y transforma los datos para su análisis.
- Una implementación de referencia para esta arquitectura está disponible en [GitHub](#).



# Hoja de referencia rápida

## (proceso de diseño de un almacenamiento de datos)



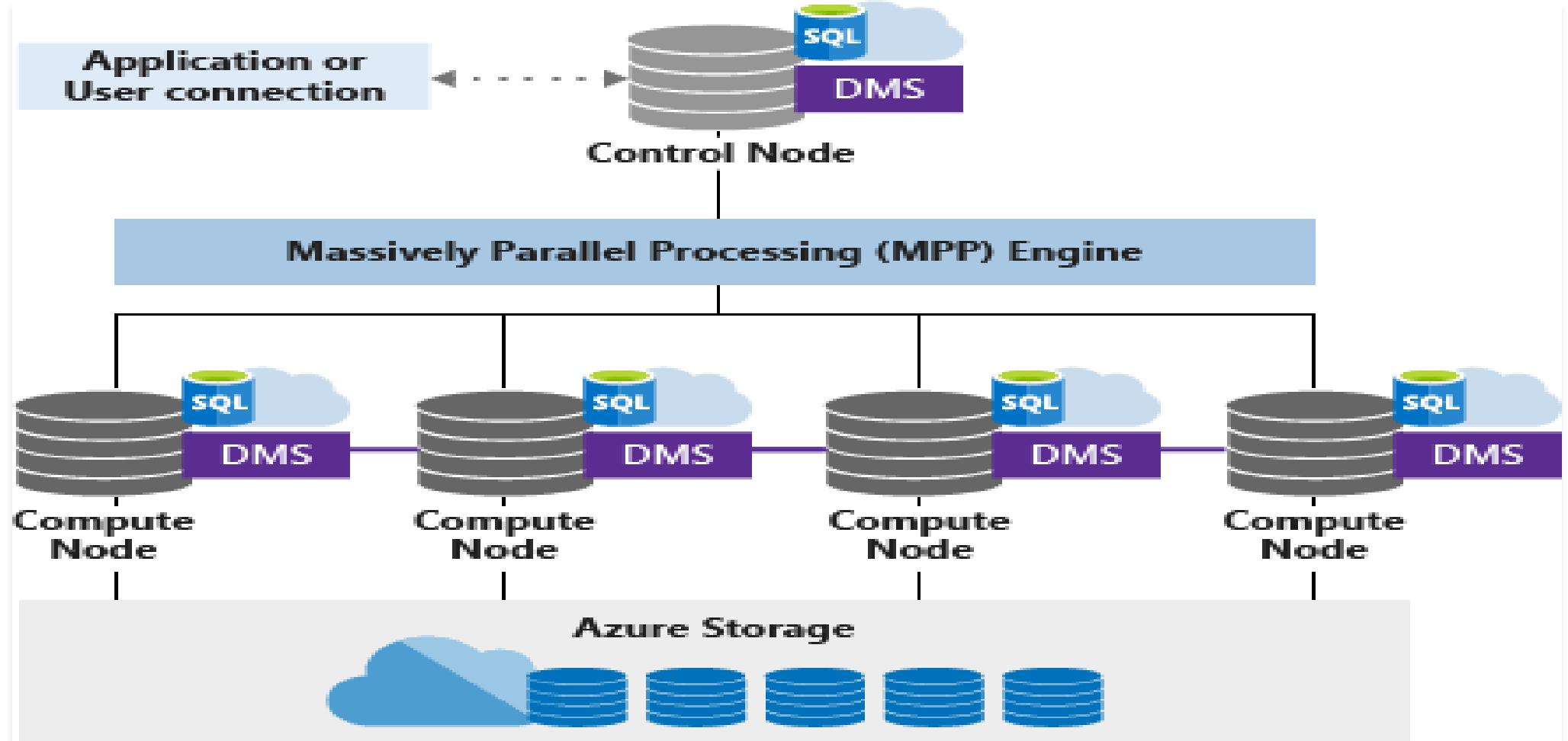
# Componentes de la arquitectura de MPP de SQL de Synapse

- SQL de Synapse aprovecha una arquitectura de escalabilidad horizontal para distribuir el procesamiento de cálculo de datos entre varios nodos. La unidad de escalado es una abstracción de la eficacia de proceso que se conoce como unidad de almacenamiento de datos. Como el proceso está separado del almacenamiento, se puede escalar con independencia de los datos del sistema.
- SQL Analytics usa una arquitectura basada en nodos. Las aplicaciones se conectan y emiten comandos a un nodo de control, que es el único punto de entrada de SQL Analytics. El nodo de control ejecuta el motor de MPP que optimiza las consultas para el procesamiento en paralelo y, después, pasa las operaciones a los nodos de ejecución para hacer su trabajo en paralelo.

# Componentes de la arquitectura de MPP de SQL de Synapse

- Los nodos de ejecución almacenan todos los datos del usuario en Azure Storage y ejecutan las consultas en paralelo. El servicio de movimiento de datos (DMS) es un servicio interno de nivel de sistema que mueve datos entre los nodos según sea necesario para ejecutar consultas en paralelo y devolver resultados precisos.
- Con el almacenamiento y el proceso desacoplados, el uso del grupo de SQL de Synapse permite:
- Cambiar la potencia del proceso independientemente de las necesidades de almacenamiento.
- Aumentar o reducir la capacidad de proceso en un grupo de SQL (almacenamiento de datos), sin mover los datos.
- Pausar la capacidad de proceso mientras se dejan los datos intactos, por lo que solo paga por el almacenamiento.
- Reanudar la capacidad de proceso durante las horas operativas.

# Componentes de la arquitectura de MPP de SQL de Synapse





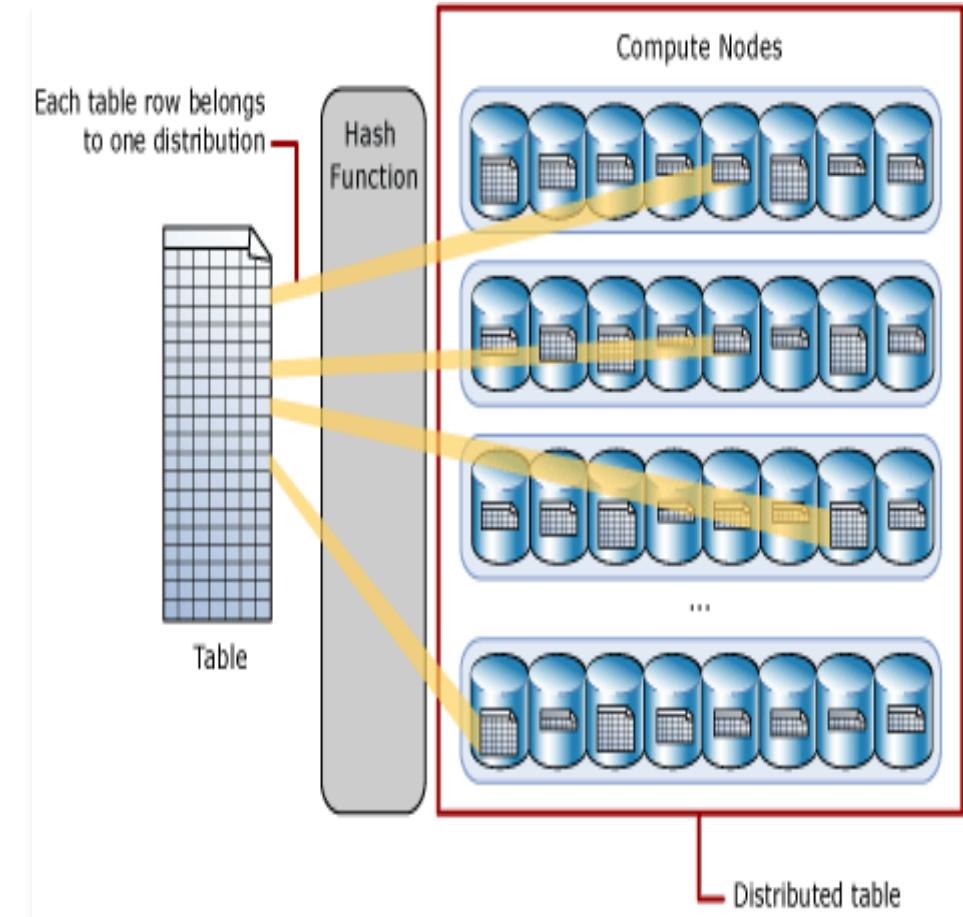
# Tablas distribuidas mediante una función hash

- Una tabla con distribución por hash puede ofrecer el máximo rendimiento de consultas para combinaciones y agregaciones en tablas grandes.
- Para particionar los datos en una tabla con distribución por hash, se usa una función hash para asignar de una manera determinista cada fila a una distribución. En la definición de tabla, una de las columnas se designa como columna de distribución. La función hash usa el valor de la columna de distribución para asignar cada fila a una distribución.
- El siguiente diagrama muestra cómo se almacena una tabla completa (no distribuida) como una tabla distribuida mediante una función hash.

# Tablas distribuidas mediante una función hash

- Cada fila pertenece a una distribución.
- Un algoritmo hash determinista asigna cada fila a una distribución.
- El número de filas de la tabla por cada distribución varía, lo que se hace patente en los diferentes tamaños de tablas.

Es preciso tener en cuenta consideraciones de rendimiento al seleccionar una columna de distribución, tales como la diferenciación, la asimetría de datos o los tipos de consultas que se ejecutan en el sistema.



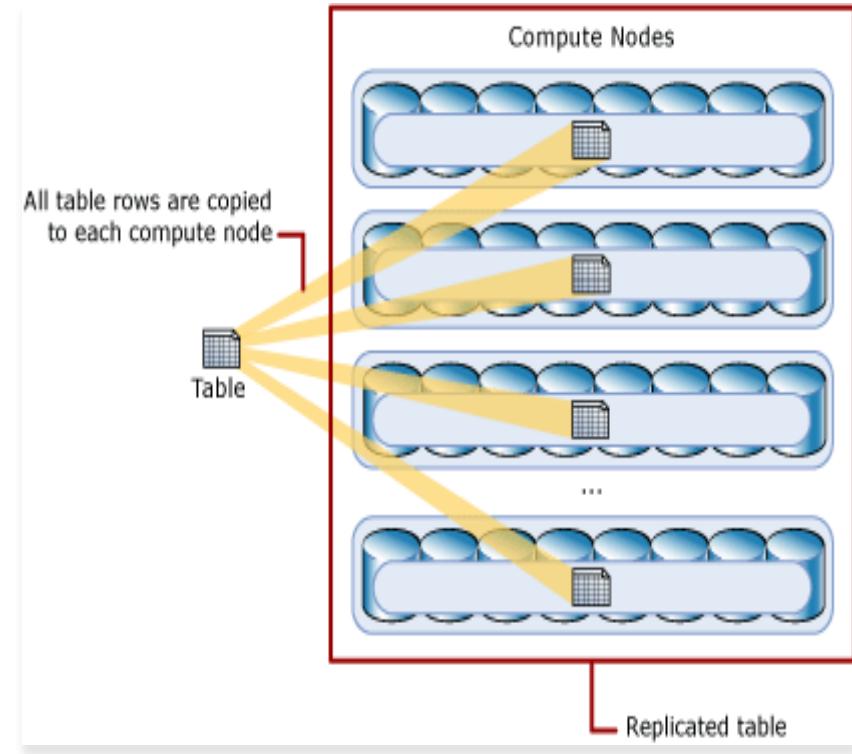


# Tablas distribuidas con el método round robin

- Una tabla round robin es la tabla más sencilla de crear y ofrece un rendimiento rápido cuando se usa como tabla de almacenamiento provisional para las cargas.
- Una tabla distribuida con el método round robin distribuye los datos uniformemente en la tabla, pero sin ninguna optimización adicional. Una distribución se elige primero de manera aleatoria y, después, los búferes de filas se asignan a las distribuciones secuencialmente. Es rápido cargar datos en una tabla round robin, pero el rendimiento de las consultas puede mejorar con tablas con distribución por hash. Las combinaciones de tablas round robin requieren reconstruir los datos, y esto requiere tiempo adicional.

# Tablas replicadas

- Una tabla replicada proporciona el rendimiento de consultas más rápido para tablas pequeñas.
- Una tabla que se replica tiene una copia completa de la tabla almacenada en la caché de cada nodo de proceso. Por lo tanto, al replicar una tabla se elimina la necesidad de transferir sus datos de un nodo de proceso a otro antes de una combinación o agregación. Las tablas replicadas se usan mejor con tablas pequeñas. Se requiere almacenamiento adicional y hay sobrecargas adicionales que se producen al escribir datos que hacen que las tablas grandes sean poco prácticas.





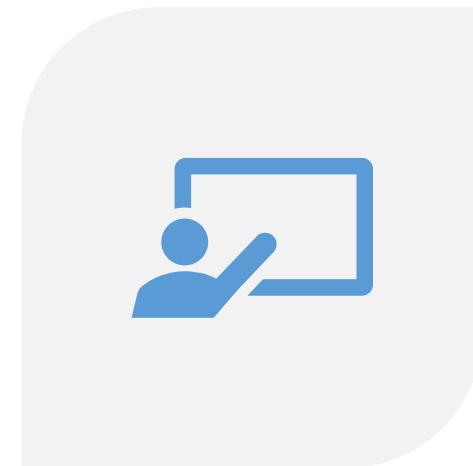
**Demo**

---

# Referencias



[DEMO 01](#)



[DEMO 02](#)

# Jesus Gil

## MVP Data Platform



@jesus\_gilv



[linkedin.com/in/jesúsgilv/](https://linkedin.com/in/jesúsgilv/)



[github.com/jesusgilv/](https://github.com/jesusgilv/)



SQLRudo



# Gracias!

