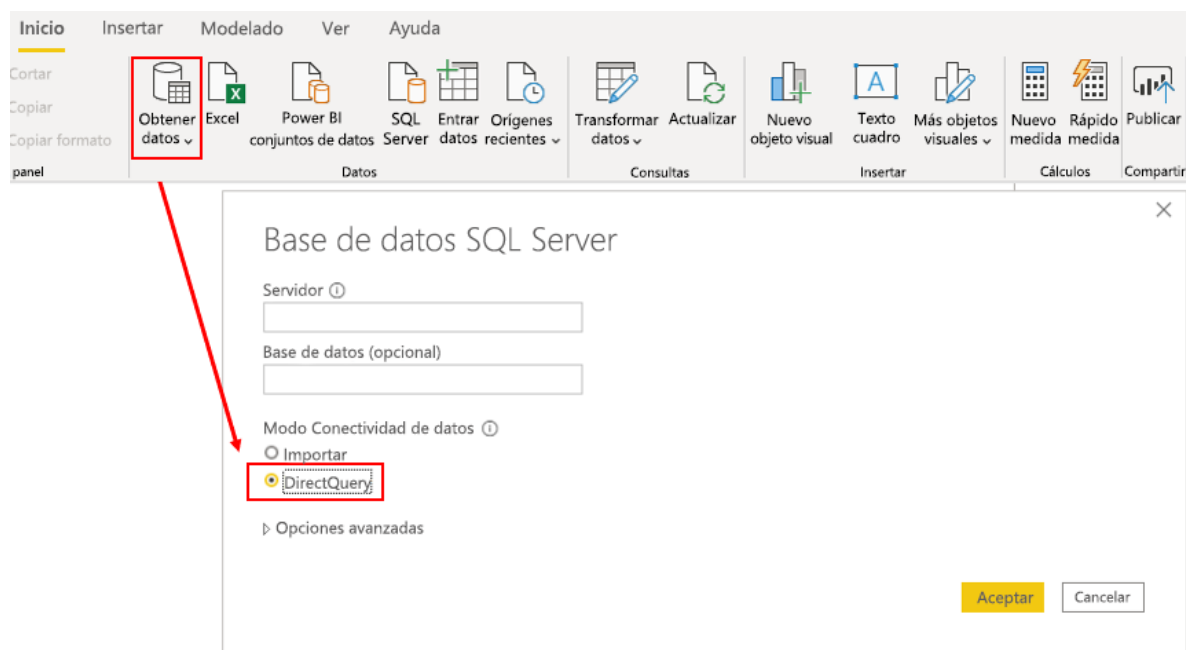


# Optimización de modelos de DirectQuery con el almacenamiento de nivel de tabla

DirectQuery es una manera de obtener datos en Power BI Desktop. El método DirectQuery implica la conexión directa a los datos en su repositorio de origen desde Power BI Desktop. Es una alternativa a la importación de datos en Power BI Desktop.



Cuando se usa el método DirectQuery, la experiencia global del usuario depende en gran medida del rendimiento del origen de datos subyacente. Los tiempos de respuesta de consulta lentos darán lugar a una experiencia del usuario negativa y, en el peor de los casos, las consultas podrían agotar el tiempo de espera. Además, el número de usuarios que abren los informes en un momento dado afectará a la carga que se coloca en el origen de datos. Por ejemplo, si el informe tiene 20 objetos visuales y lo usan 10 personas, habrá 200 consultas o más en el origen de datos, ya que cada objeto visual emitirá una o más.

Desafortunadamente, el rendimiento del modelo de Power BI no solo se verá afectado por el del origen de datos subyacente, sino también por otros factores que no se pueden controlar, como los siguientes:

- Latencia de red; las redes más rápidas devuelven datos con más rapidez.
- El rendimiento del servidor del origen de datos y la cantidad de cargas de trabajo adicionales en ese servidor. Por ejemplo, considere las implicaciones de una

actualización de servidor que se realiza mientras cientos de usuarios utilizan el mismo servidor por distintos motivos.

Por tanto, el uso de DirectQuery supone un riesgo para la calidad del rendimiento del modelo. Para optimizar el rendimiento en esta situación, debe tener control sobre la base de datos de origen, o tener acceso a ella.

Para obtener información más detallada, vea [Instrucciones del modelo de DirectQuery en Power BI Desktop](#).

## Implicaciones del uso de DirectQuery

Un procedimiento recomendado consiste en importar los datos en Power BI Desktop, pero es posible que la organización necesite usar el modo de conectividad de datos de DirectQuery por uno de los motivos siguientes (ventajas de DirectQuery):

- Es adecuado cuando los datos cambian con frecuencia y se necesitan crear informes en tiempo real.
- Puede controlar datos de gran tamaño sin necesidad de agregarlos previamente.
- Aplica las restricciones de soberanía de datos para cumplir con los requisitos legales.
- Se puede usar con un origen de datos multidimensional que contenga medidas como SAP Business Warehouse (BW).

Si la organización tiene que usar DirectQuery, debe entender claramente su comportamiento en Power BI Desktop y tener en cuenta sus limitaciones. De ese modo, estará en la posición correcta para adoptar medidas a fin de optimizar el modelo de DirectQuery lo máximo posible.

## Comportamiento de las conexiones de DirectQuery

Cuando se usa DirectQuery para conectarse a los datos de Power BI Desktop, la conexión se comporta de esta forma:

- Al usar inicialmente la característica **Obtener datos** de Power BI Desktop, se selecciona el origen. Si se conecta a un origen relacional, puede seleccionar un conjunto de tablas y cada una de ellas definirá una consulta que devuelve un conjunto de datos de forma lógica. Si selecciona un origen multidimensional, como SAP BW, solo puede seleccionar el origen.
- Al cargar los datos, no se importan en Power BI Desktop; solo se carga el esquema. Al crear un objeto visual dentro de Power BI Desktop, se envían consultas al origen subyacente para recuperar los datos necesarios. El tiempo que se tarda en actualizar el objeto visual depende del rendimiento del origen de datos subyacente.
- Si se realizan cambios en los datos subyacentes, no se reflejarán inmediatamente en los objetos visuales existentes en Power BI debido al almacenamiento en caché. Tendrá que realizar una actualización para ver esos cambios. Las consultas necesarias se reenvían para cada uno de los objetos visuales, que se actualizan en consecuencia.
- Al publicar el informe en el servicio Power BI, se genera un conjunto de datos, como sucede con la importación. Pero con ese conjunto de datos no se incluye ningún dato.
- Cuando se abre un informe existente en el servicio Power BI (o se crea uno) se vuelve a consultar el origen subyacente para recuperar los datos necesarios. En función de la ubicación del origen original, es posible que tenga que configurar una puerta de enlace de datos local.
- Puede anclar objetos visuales, o páginas de informe completas, como iconos del panel. Los iconos se actualizan de forma automática según una programación, por ejemplo, cada hora. Puede controlar la frecuencia de esta actualización para ajustarla a sus necesidades. Al abrir un panel, los iconos reflejan los datos en el momento de la última actualización y es posible que no incluyan los últimos cambios realizados en el origen de datos subyacente. Siempre puede actualizar un panel abierto para asegurarse de que esté actualizado.

## Limitaciones de las conexiones de DirectQuery

El uso de DirectQuery puede tener implicaciones negativas. Las limitaciones varían en función del origen de datos específico que se use. Debe tener en cuenta los aspectos siguientes:

- **Rendimiento:** como se ha indicado antes, la experiencia global del usuario depende en gran medida del rendimiento del origen de datos subyacente.
- **Seguridad:** si usa varios orígenes de datos en un modelo de DirectQuery, es importante comprender cómo se mueven los datos entre los orígenes de datos

subyacentes y las implicaciones de seguridad asociadas. También debe identificar si se aplican reglas de seguridad a los datos del origen subyacente porque, en Power BI todos los usuarios pueden ver los datos.

- **Transformación de datos:** en comparación con los datos importados, los datos que se originen desde DirectQuery tienen limitaciones en cuanto a la aplicación de técnicas de transformación de datos en el Editor de Power Query. Por ejemplo, si se conecta a un origen OLAP, como SAP BW, no puede hacer ninguna transformación; se toma el modelo externo completo del origen de datos. Si quiere realizar cualquier transformación en los datos, tendrá que hacerlo en el origen de datos subyacente.
- **Modelado:** algunas de las funcionalidades de modelado que tiene con los datos importados no están disponibles, o están limitadas, cuando se usa DirectQuery.
- **Generación de informes:** casi todas las funcionalidades de generación de informes que tiene con los datos importados también se admiten para los modelos de DirectQuery, siempre que el origen subyacente ofrezca un nivel de rendimiento adecuado. Pero cuando el informe se publica en el servicio Power BI, no se admiten las características Conclusiones rápidas y Preguntas y respuestas. Además, el uso de la característica Explorar en Excel probablemente generará un rendimiento más bajo.

Para obtener información más detallada sobre las limitaciones del uso de DirectQuery, vea [Implicaciones del uso de DirectQuery](#).

Ahora que tiene una breve comprensión de cómo funciona DirectQuery y las limitaciones que plantea, puede adoptar medidas para mejorar el rendimiento.

## Optimización del rendimiento

Siguiendo con el escenario de Tailwind Traders, durante la revisión del modelo de datos, descubre que la consulta ha usado DirectQuery para conectar Power BI Desktop a los datos de origen. Este uso de DirectQuery es el motivo por el que los usuarios experimentan un rendimiento deficiente del informe. La carga de las páginas en el informe tarda demasiado y las tablas no se actualizan con la rapidez suficiente cuando se realizan determinadas selecciones. Tendrá que adoptar medidas para optimizar el rendimiento del modelo de DirectQuery.

Puede examinar las consultas que se envían al origen subyacente e intentar identificar el motivo de su bajo rendimiento. Después, puede realizar cambios en Power BI Desktop y en el origen de datos subyacente para optimizar el rendimiento general.

## Optimización de datos en Power BI Desktop

Cuando haya optimizado el origen de datos tanto como sea posible, puede realizar más acciones en Power BI Desktop mediante el **Analizador de rendimiento**, donde puede aislar las consultas para validar los planes de consulta.

Puede analizar la duración de las consultas que se envían al origen subyacente para identificar las que tardan mucho tiempo en cargarse. En otras palabras, puede identificar dónde se producen cuellos de botella.

No es necesario usar un enfoque especial al optimizar un modelo de DirectQuery; puede aplicar las mismas técnicas de optimización que ha usado en los datos importados para optimizar los datos del origen de DirectQuery. Por ejemplo, puede reducir el número de objetos visuales en la página del informe o el número de campos que se usan en un objeto visual. También puede quitar las columnas y filas innecesarias.

Para obtener instrucciones más detalladas sobre cómo optimizar una consulta de DirectQuery, vea: [Instrucciones del modelo de DirectQuery en Power BI Desktop](#) e [Instrucciones para usar DirectQuery correctamente](#).

## Optimización del origen de datos subyacente (base de datos conectada)

La primera parada es el origen de datos. Tendrá que optimizar la base de datos de origen tanto como sea posible, ya que todo lo que haga para mejorar su rendimiento mejorará también DirectQuery en Power BI. Las acciones que realice en la base de datos serán las más beneficiosas.

Considere la posibilidad de usar los siguientes procedimientos estándar de base de datos que se aplican a la mayoría de las situaciones:

- Evite el uso de columnas calculadas complejas porque la expresión de cálculo se insertará en las consultas de origen. Es más eficaz volver a insertar la expresión en el origen porque evita la desactivación. También puede considerar la posibilidad de agregar columnas de clave suplente a tablas de tipo de dimensión.
- Revise los índices y compruebe que la indexación actual es correcta. Si tiene que crear índices, asegúrese de que sean adecuados.

Consulte los documentos de instrucciones del origen de datos e implemente sus recomendaciones de rendimiento.

## Personalización de las opciones de reducción de consulta

Power BI Desktop le ofrece la posibilidad de enviar menos consultas y de deshabilitar determinadas interacciones que darán lugar a una experiencia deficiente si las consultas resultantes tardan mucho tiempo en ejecutarse. Al aplicar estas opciones, se evita que las consultas visiten repetidamente el origen de datos, lo que mejora el rendimiento.

En este ejemplo, modificará la configuración predeterminada para aplicar las opciones de reducción de datos disponibles al modelo. Para acceder a la configuración, seleccione **Archivo > Opciones y configuración > Opciones**, desplácese hacia abajo en la página y, después, seleccione la opción **Reducción de consulta**.

Están disponibles las siguientes opciones de reducción de consulta:

- **Reducción del número de consultas enviadas:** de forma predeterminada, cada objeto visual interactúa con todos los demás objetos visuales. Al activar esta casilla, se deshabilita la interacción predeterminada. Opcionalmente, puede elegir qué objetos visuales interactúan entre sí mediante la característica **Editar interacciones**.
- **Segmentaciones:** la opción **Aplicar cambios en la segmentación al instante** está seleccionada de manera predeterminada. Para forzar que los usuarios del informe apliquen manualmente los cambios de la segmentación, seleccione la opción **Agregue un botón Aplicar a cada segmentación para aplicar los cambios cuando esté a punto**.
- **Filtros:** la opción **Aplicar los cambios del filtro básico al instante** está seleccionada de manera predeterminada. Para forzar a los usuarios del informe a que apliquen manualmente los cambios de filtro, seleccione una de las opciones alternativas:
  - **Agregar botones Aplicar a todos los filtros básicos para aplicar los cambios cuando esté listo**
  - **Agregar un único botón Aplicar al panel de filtros para aplicar los cambios a la vez (versión preliminar)**

