



Tratamiento de datos con **Power Query**

Ana María Bisbé York

Prólogo de Miguel Llopis

Principal Group Program Manager, Microsoft Data Integration

Índice de contenidos

Prólogo de Miguel Llopis.....	16
Cómo usar este libro.....	20
1. Introducción a la transformación y limpieza de datos.....	26
Los datos en el mundo de hoy	27
Desafíos que abordar	27
Conocer los datos y el objeto de estudio	28
Comprobar de la calidad del dato	28
Afrontar las diferentes formas de presentación original de los datos.....	30
Comprender la distribución de la muestra	31
Valorar la necesidad de segmentación.....	32
Dar tratamiento de valores atípicos	32
Identificar escenarios de cuantificación.....	33
Optimizar los tiempos de carga y actualización de datos.....	33
Preparar datos para el análisis	34
Conclusiones.....	35
2. Introducción a Power Query.....	36
¿Qué es Power Query?	37
¿Para qué sirve Power Query?	38
Tipo de tareas que se realizan en Power Query	39
¿Dónde se encuentra Power Query?	42
Microsoft Excel	42
Power BI Desktop	42
Power BI DataFlows.....	43
Otras herramientas y entornos	43
¿Qué elementos componen Power Query?	44
Estructura.....	44
Interfaz de usuario.....	45

Tipos de datos.....	53	6. Obtener datos externos de web	116
Tipos de objetos.....	55	Formas de acceso a datos en orígenes externos vía web.....	117
Tipos de consultas.....	57	Retos al consumir datos de la web.....	117
El lenguaje M como motor.....	58	Estructura del dato	117
Conclusiones.....	61	Autoría de contenidos.....	117
3. Introducción al modelo tabular.....	62	Importar datos desde un origen web a Power Query	118
¿Qué es el modelo tabular?.....	63	Navegador para el conector web.....	118
Modelo tabular en Power Pivot de Microsoft Excel	63	Privacidad en conexión web.....	120
Modelo tabular en Power BI Desktop.....	65	Consulta de conexión web.....	121
Tipos de tablas en el modelo y sus características.....	66	Crear tablas a partir del ejemplo.....	121
Otros conceptos del modelo tabular	67	Ejemplo 1. Todos los datos: dos columnas País y Sitio.....	122
El perfil de distribución y el modelo tabular	68	Ejemplo 2. Todos los países y solo el primer sitio de cada uno.....	123
Papel de las relaciones del modelo tabular	70	Importar un archivo desde web.....	124
¿Qué tipos de esquemas existen en el modelo tabular?	77	Descargar y luego conectar.....	125
Tabla única	77	Acceder a información web utilizando API	126
Esquema copo de nieve.....	78	¿Qué es API?	126
Esquema estrella.....	80	¿Cómo se obtienen las claves?.....	126
Solución a problemas y escenarios de modelado.....	81	Ejemplo de uso	127
¿Qué son las entidades y los atributos del modelo?	82	Acceder a Google Sheets	128
Entidad	82	Obtener el conector.....	129
Atributo	83	Iniciar sesión.....	129
¿Cómo se puede examinar el modelo tabular?.....	85	Conectar con datos abiertos OData	130
Herramientas externas	85	Vista en un navegador web	131
Tabular Editor.....	88	Conexión con Power Query	131
Conclusiones.....	89	Ajustar la consulta aplicando filtros.....	131
4. Obtener datos externos de archivos.....	90	Conectar a datos con formato Parquet.....	132
Tareas de extracción.....	91	Conectar con contenido en Sharepoint	133
Conectar a archivos planos.....	91	Power Query en Excel y conectores web	133
Configuración regional y configuración de Windows	93	Conclusiones.....	133
Diferenciar entre archivos delimitados y no delimitados.....	93	7. Otros orígenes de datos.....	134
Conectar a libros Excel.....	94	El Servicio Power BI como origen de datos.....	135
Formatos.....	95	Conjunto de datos en Power BI	135
Desde Excel a Power BI y de Power a Excel	95	Power BI DataFlows	138
Importar desde Excel a Power BI	96	Power BI Datamarts	139
Conectar a PDF	98	Azure.....	140
Importar desde archivo JSON	98	Azure Analysis Services.....	140
Importar desde XML.....	98	Microsoft Exchange.....	141
Importar desde carpetas.....	98	Conclusiones.....	143
Conclusiones.....	101	8. Configurar opciones de acceso a datos	144
5. Obtener datos externos de bases de datos.....	102	Especificar datos en consultas desconectadas	145
Conectar a base de datos SQL Server desde Power Query	103	Reutilizar conexiones existentes	146
Plegado de consultas	106	Configurar propiedades de orígenes de datos.....	147
Modos de conexión	111	Paso Origen en la consulta.....	147
Importar	111	Barra de fórmulas.....	148
DirectQuery	112	Configuración de origen de datos	148
Modo de almacenamiento.....	115	Trabajar con parámetros de ruta y propiedades de conexión.....	150
Conclusiones.....	115	Parámetros de ruta.....	150
		Parámetros de filtro o validación.....	150

Parámetros de entorno vs. parámetros de consulta.....	151
Editar valores de parámetros desde Power BI Desktop.....	153
Editar valores de parámetros desde Servicio Power BI.....	153
Examinar dependencias de consultas para distinguir rutas de acceso.....	154
Agrupar elementos en el editor de consultas Power Query.....	156
Describir los procesos con comentarios y modificar los nombres de los pasos.....	157
Puertas de enlace, el puente a los datos locales.....	158
¿Qué es una puerta de enlace?.....	158
¿Qué tipos de puertas de enlace existen?.....	158
¿Cómo se instala una puerta de enlace?.....	158
Conclusiones.....	159
9. Explorar contenido de las columnas.....	160
Significado de los colores en el perfil de calidad de datos.....	161
Identificar y asignar tipos de datos.....	163
Identificar el momento adecuado para cambiar el tipo de datos.....	164
Distinguir el contenido de la muestra con el perfil de distribución.....	164
Detectar detalles en los datos activando el perfil de columnas.....	166
Más sobre los perfiles de datos.....	168
Obtener perfil de datos con información básica.....	168
Obtener perfil de datos con información adicional.....	169
Explorar las acciones de filtro por filas.....	170
Filtros y expresiones en lenguaje M.....	170
Filtrar múltiples columnas.....	173
Filtrar con Uso avanzado.....	173
Comprobar eficacia de nombres de columnas y tablas.....	174
Conclusiones.....	175
10. Comprobar y garantizar la calidad de los datos.....	176
Importancia de velar por la calidad de datos.....	177
Importancia de velar por la calidad del modelo.....	177
Sobre las transformaciones.....	178
Limpiar cadenas indicando mayúsculas o minúsculas.....	178
Limpiar eliminando filas y columnas.....	179
Eliminar filas con valores en blanco o nulos.....	179
Eliminar columnas innecesarias.....	181
Eliminar espacios para mejorar la calidad.....	182
Null vs. <i>null</i> vs. vacío.....	183
Reemplazar vs. columna condicional.....	185
Eliminar valores nulos rellenando por columnas.....	185
Quitar duplicados para garantizar la sostenibilidad del dato.....	186
Cambiar tipo de datos con configuración local.....	187
Conclusiones.....	189
11. Transformar y limpiar columnas de texto.....	190
Extraer parte de una cadena de texto.....	191
Extraer contenido entre delimitadores.....	191
Extraer vs. reemplazar elementos.....	191
Diferencias entre Recortar y Limpiar.....	194
Función Recortar.....	195
Función Limpiar.....	196
Extraer números de textos.....	198
Posibles errores en columnas de texto.....	199
Agrupar por columna con errores.....	200
Filtrar desde columna con errores.....	201
Calidad de datos y perfil de distribución.....	202
Evaluar condiciones sobre cadenas de texto sin limpiar.....	202
Conclusiones.....	203
12. Transformar y limpiar columnas tipo Fecha.....	204
Escenarios de conversión de datos tipo Fecha.....	205
Problemas de conversión según la configuración regional.....	205
Formatos que se convierten automáticamente.....	206
Formatos que no se convierten y hay que transformar.....	207
Crear tabla de fechas o calendario en Power Query.....	209
Especificar datos para crear tabla de fechas.....	210
Crear tabla fechas con expresiones en lenguaje M.....	210
Agregar atributos desde el menú.....	214
Agregar atributos desde la ayuda más allá de la interfaz.....	217
Crear una consulta con las referencias del lenguaje M.....	218
Crear una función personalizada a partir de la referencia integrada.....	218
Utilizar una función personalizada para crear nuevos atributos a la tabla calendario.....	221
Atributos de calendario y el modelo de datos.....	222
Conclusiones.....	223
13. Transformaciones para cambiar la estructura de consultas.....	224
Combinar columnas.....	225
Desventajas de combinar columnas.....	227
Insertar columna de Índice en lugar de combinar.....	227
Dividir columnas.....	228
Dividir columnas para estructurar mejor su contenido.....	228
Dividir columnas controlando el uso de caracteres y delimitadores.....	230
Dividir texto en filas en Power Query.....	232
Acción adicional si el delimitador es un espacio.....	232
Dividir en filas y combinar columnas.....	233
Transponer filas y columnas.....	235
Transponer para rellenar.....	236
Agrupar filas.....	236
Resultado tras agrupar filas en una consulta.....	237
Agrupar y ver el detalle.....	238
Agrupar más allá de la propuesta de la interfaz.....	238
Agrupar y concatenar el resultado.....	240
Dinamizar columnas en Power Query.....	240
Dinamizar columnas y crear agregados.....	242
Caso de error al dinamizar columnas.....	243
Solución al error al dinamizar columnas.....	243
Dinamizar columnas agregando un índice.....	244
Anular dinamización de columnas o Unpivot.....	246
Anular dinamización de columnas y celdas vacías.....	248
Conclusiones.....	249

14. Enriquecer modelo con nuevas columnas	250		
Agregar columnas necesarias para un informe analítico	251		
Diferenciar casos para transformar o agregar columna	252		
Evaluar la composición, calidad y distribución de los valores en una columna	253		
Trabajar con columnas condicionales	254		
Comparar valores y crear una columna condicional	255		
Los nulos y la evaluación en columnas condicionales	256		
Al preguntar por valores nulos el orden importa	257		
Sustituir valores de una columna para eliminar valores innecesarios	258		
Las columnas condicionales y el orden de las evaluaciones	258		
Obtener nuevos atributos	260		
Crear nuevas columnas asignando valores numéricos	260		
Para definir el orden de visualización de las columnas	261		
Para cuantificar las cadenas de texto	261		
Aprovechar los recursos de la opción Columnas desde el ejemplo	262		
Mostrar segmentación en rangos equidistantes	263		
Combinar columnas, empleando todo o parte del contenido	264		
Cuando parece que se ha interpretado bien, pero no es así	266		
Conclusiones	267		
15. Combinar consultas	268		
Referenciar y duplicar para obtener nuevas consultas	269		
Referenciar	269		
Duplicar	270		
Otras acciones vs. Duplicar y referenciar	271		
Dependencias de consultas	272		
Bases para combinar consultas	273		
Combinar consultas y el origen de datos relacional	273		
Elementos que componen la combinación de consultas	274		
Tipos de uniones en Power Query	275		
Desafíos en el tratamiento de textos y la combinación de consultas	275		
Combinación con coincidencia aproximada (<i>fuzzy</i>) para columnas de texto	276		
Combinación aproximada con tabla de equivalencias	277		
Definir el umbral de similitud	278		
La combinación de consultas y los niveles de privacidad	278		
Ver la configuración de los permisos	279		
Otros ejemplos de uso	279		
Detectar cambios en los datos comparando consultas	280		
Obtener el valor real a partir de valores acumulados	281		
Conclusiones	283		
16. El modelo tabular y el rol de Power Query	284		
Pasar de tabla única a modelo en estrella	285		
Pasar de copo de nieve a modelo en estrella	287		
Definir entidades	288		
Definir atributos del modelo	289		
Desde tabla cruzada o columna apilada	289		
Cuando no existen los atributos	290		
Eliminar complejidad creando entidades misceláneas	292		
Crear consulta para Misceláneas desde tabla única	293		
Crear consulta para Misceláneas desde producto cartesiano entre tablas no relacionadas	295		
Crear entidad única anexando consultas	297		
Anexar datos desde orígenes desconectados o archivos planos	297		
Anexar datos desde hojas de Excel	298		
Modelar escenarios de entidades con múltiples roles	301		
Variar dimensiones diferenciadas por roles o funciones	302		
Dimensión única realizadora de múltiples funciones (RPD)	303		
Crear entidades para uso compartido	305		
Dos tablas de hechos vs. tabla única	306		
Dimensiones compartidas y cardinalidad de la relación	307		
Dimensión compartida entre dimensiones	307		
El modelado de tabla de tiempos dentro y fuera de Power Query	308		
Configurar Inteligencia de tiempo en Power BI Desktop	309		
Tablas de fechas ocultas en el modelo de datos	309		
Contenido de las tablas ocultas	310		
Eliminar las tablas de fecha ocultas en el modelo	311		
Agrupar y agregar vs. mantener detalle, pros y contras	312		
Agrupar y agregar en escenarios de cabecera y detalle	313		
Cuando se detallan fechas, horas y minutos	314		
Proteger la estructura de la tabla de hechos	317		
Solución para tabla de hechos	318		
Conclusiones	319		
17. Solucionar escenarios de modelado de relaciones con Power Query	320		
Evitar relación 1 a 1 entre dos tablas de hechos	321		
Evitar relación 1 a 1 entre dimensiones	324		
Evitar relación M a M entre dos tablas quitando duplicados	324		
Evitar relación M a M entre dos tablas quitando nulos	326		
Evitar relación M a M entre dos tablas con una tabla puente entre hechos	327		
Evitar relación M a M entre dos tablas con una tabla puente entre hechos y dimensiones	328		
Dar solución a escenarios con relación variable entre columnas	330		
Evitar filas huérfanas en los hechos por errores en la integridad referencial	333		
Conclusiones	335		
18. Distinguir y controlar errores en Power Query	336		
Prevención de errores	337		
Tipos de errores detectados en consultas de Power Query	337		
Error de nivel de paso	337		
Error a nivel de celda o fila	339		
El perfil de calidad de columna y los errores	340		
Errores y tipos de datos	341		
Errores de operación	342		
Evaluar una condición con el tipo de datos erróneo	342		
Evitar error de navegación al combinar archivos	343		
Control de errores	344		
Función TRY	345		
Función TRY sin argumentos adicionales	345		
Función TRY y sus cláusulas otherwise y catch	347		

Errores en la carga de datos.....	349
Ignorar el mensaje de error en la carga.....	349
Ver las filas con error en Power Query	350
Eliminar los errores antes de cargar	351
Conclusiones.....	351
19. Buenas prácticas en la creación de consultas con Power Query ..	352
Buenas prácticas	353
No ordenar por columnas.....	353
No ordenar por filas	354
Elegir cuándo definir el tipo de datos	354
Aplicar mejoras a un grupo de columnas en lugar de ir transformando una a una	357
Ajustar tipo de datos directamente con lenguaje M.....	360
Deshabilitar la carga de una consulta	361
Trabajar con carpetas o grupos para organizar el Navegador de consultas.....	363
Asignar propiedades y comentarios a los pasos.....	364
Implementar los cambios o cargar los datos al modelo tabular desde Power Query.....	366
Conclusiones.....	367
20. Introducción a lenguaje M	368
El código M y la barra de fórmulas	370
Presentación del Editor avanzado de Power Query	370
Las palabras reservadas let e in	371
El código M y los comentarios	373
Crear consultas en blanco en Power Query	373
Trabajar con la ayuda de referencias del lenguaje M integrada en Power Query	374
Trabajar con funciones.....	375
Función if.....	376
Funciones de conversión de datos.....	378
Crear funciones personalizadas	378
Uso de función personalizada para limpiar cabeceras de columnas	379
Crear funciones propias desde la referencia del lenguaje y sin programar	380
Distinguir las secciones en el editor de consultas	381
Más allá de #shared y #sections	382
Estructuras en Power Query.....	383
Las listas como objetos en el lenguaje M	383
Crear lista desde el menú.....	386
Listas y funciones.....	386
Lista consecutiva con rango	387
Los registros como objetos del lenguaje M.....	387
Utilizar o no la capacidad Intellisense o Autocompletar	389
Obtener un registro navegando por una "condición"	389
Crear un registro para cada fila de una tabla.....	390
La combinación de registros y los nombres de columnas.....	390
Las tablas como objetos del lenguaje M.....	391
Acceso a elementos de tablas	392
¿Qué es la proyección?	393
Casos de uso en escenarios puntuales.....	394
Eliminar columnas en dependencia de su posición.....	394
Prevenir errores con parámetros en funciones M	394
Calcular la edad desde la interfaz o en un único paso desde el código.....	395
Conclusiones.....	396

21. Optimización de consultas con Power Query	398
Herramienta de medición para consultas.....	399
Importancia de la herramienta.....	400
Configurar la herramienta	400
Iniciar el diagnóstico.....	400
Explorar el resultado del diagnóstico en Power Query	402
Realizar el diagnóstico a la consulta eficiente para comparar los resultados	403
Combinar consultas para analizar los resultados en Power Query	404
Explorar el resultado del diagnóstico en informes de Power BI.....	405
Medición de rendimiento que depende del plegado de consulta.....	407
Validar el plegado de consulta con M.....	408
Evaluación diferida o perezosa.....	408
Medir el rendimiento en origen con SQL Server Profiler.....	409
Evitar el uso del plegado de consulta	410
Medir el efecto del plegado de consultas	410
Otras herramientas para explorar el modelo de datos	412
Conclusiones.....	413
22. Los flujos de datos o Power Query en la nube	414
Ventajas de uso de los flujos de datos	415
¿Cómo acceder a los flujos de datos?	416
Conectar a datos externos desde flujos de datos	418
Las puertas de enlace en los flujos de datos.....	418
Vías para conectar a orígenes externos	419
Entorno de Power Query en el Servicio Power BI.....	421
Vista de esquema.....	422
Vista de diagrama	423
Plan de consulta	426
Marcador visual del tipo de combinación.....	428
Contador de ejecución.....	430
Script de la consulta	430
Control visual del plegado de consultas	431
Perfiles de columna	432
Modificar un flujo de datos	432
Otras características y novedades de los flujos de datos.....	433
Opciones del proyecto.....	433
Transformaciones en Power Query	434
Marcar columna como clave.....	434
Sobre los tipos de datos Cualquiera y Binario.....	435
Tablas calculadas en flujos de datos.....	435
Flujos de datos Gen2.....	436
Comenzar a trabajar con flujos de datos Gen2	437
Abrir Power Query en Data Factory	438
Entorno de trabajo de Power Query para flujos de datos Gen2	438
Resumen de conceptos y componentes.....	440
Conclusiones.....	441

Índice alfabético.....	442
-------------------------------	------------

1

Introducción a la transformación y limpieza de datos

En este capítulo aprenderá a:

- Valorar la importancia de los datos en el contexto actual.
- Conocer los desafíos que enfrentan los analistas de datos.
- Identificar acciones a realizar en la etapa de preparación de datos.

Los datos en el mundo de hoy

Nuestro mundo está regido por los datos, a cada paso, a cada acción, más y más datos. Son muchos y muy variados los datos, que proceden de fuentes diversas y tienen estructuras disímiles. Los datos son la base del conocimiento, de la gestión y la llamada a la acción para la economía y la vida de empresas, instituciones y personas.

No hay duda de la relevancia del tratamiento, análisis y presentación de los datos en el contexto actual.

El proceso de creación de informes, visualizaciones, cuadros de mando o historias empieza con la obtención de datos.

Se requiere seleccionar orígenes de datos que, además de ser fiables, se adapten mejor a las necesidades del análisis entre todas las fuentes de datos de que se dispone. Es preciso tener en cuenta la viabilidad de acceso a los datos, así como la posibilidad de mantenerlos estables y actualizados.

El profundo conocimiento de los datos y del caso de uso o negocio que se analiza permite definir la fuente a utilizar, el proceso de transformación a desarrollar y cómo se deben mostrar para encontrar en cada caso el beneficio de utilizar:

- Datos resumidos o detallados.
- Muestras comparativas, con relaciones o distribuciones.
- Visualizaciones para distinguir la parte del todo.

Si se cuenta con los datos de los hechos en una serie temporal de calidad, se podrá identificar escenarios para análisis en línea de tiempo que evidencian evolución y tendencias. Los datos que muestran información geolocalizada de calidad facilitan exponer los resultados utilizando mapas de forma eficiente.

Para crear informes con buena calidad, hay que partir de la base de que los datos han de estar limpios y bien preparados. Sin ello, se dificulta o simplemente no se consigue alcanzar resultados adecuados, por lo que es de vital importancia entender el gran valor que aporta la gestión de datos en las organizaciones en el mundo de hoy.

Desafíos que abordar

La tarea de transformación y limpieza de datos implica abordar múltiples desafíos. Lo más habitual es que la persona que se dedica al análisis de datos se nutra de datos de orígenes externos que no le pertenecen, por lo que se adapta a lo que recibe, cuando lo recibe y en el estado que lo recibe.

La figura 2.6 muestra todos estos elementos que integran la interfaz de usuario o entorno de trabajo en Power Query integrado en Power BI Desktop.

Menús y cintas de opciones

El editor de consultas tiene una banda en la parte superior con pestañas o fichas que contienen grupos de opciones cuyo objetivo es facilitar la manipulación y limpieza de datos, especialmente en la etapa inicial. La organización por funciones y tipos de datos es muy acertada.

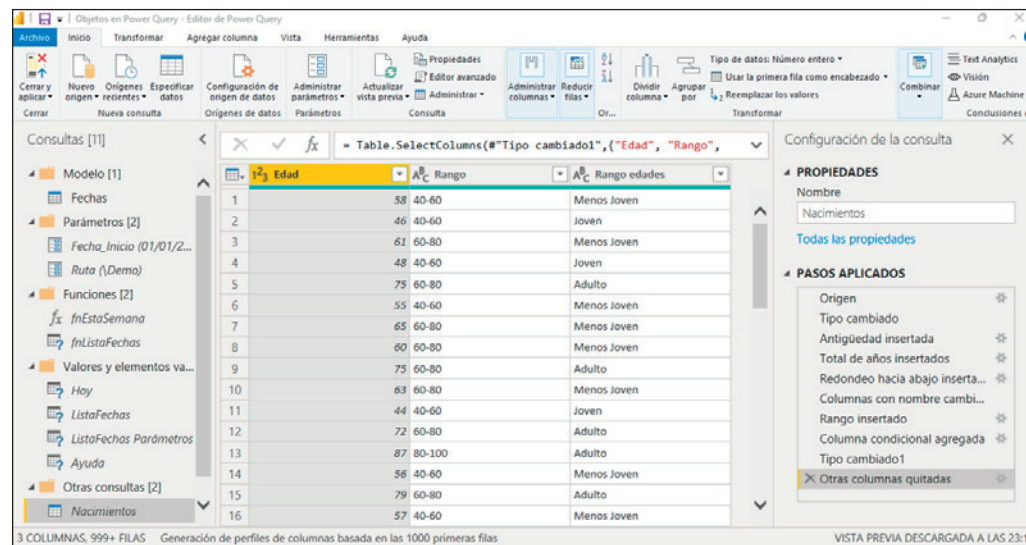


Figura 2.6. Interfaz de usuario de Power Query en Power BI Desktop.

La figura 2.7 muestra un fragmento de los grupos y las opciones disponibles en el menú Inicio.

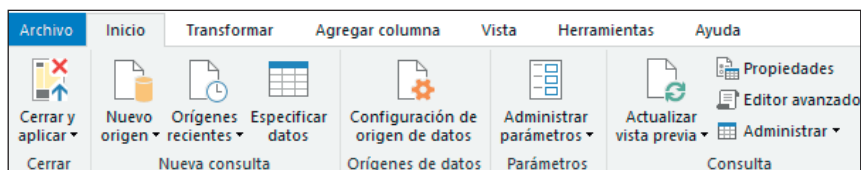


Figura 2.7. Fragmento de la cinta de opciones del editor de consultas.

Panel de navegación de Consultas

Se trata de un panel lateral, situado a la izquierda, que incluye todos los elementos con los que se trabaja en cada caso. Este panel se puede contraer para ganar espacio en la consulta.

Cada tipo de elemento cuenta con un menú contextual propio. La figura 2.8 muestra un ejemplo de panel de navegación que incluye objetos de tipos diferentes agrupados en carpetas.

Se sugiere agrupar elementos en carpetas teniendo en cuenta los siguientes apuntes:

- Los elementos se pueden y deben agrupar en carpetas para su rápida localización.
- El nombre de cada carpeta se corresponde con el nombre del grupo y se acompaña de la cantidad de elementos que integran el grupo.
- Todos los nombres de grupos se pueden cambiar, salvo el nombre Otras consultas, que es creado automáticamente por Power Query a la vez que se crea el primer grupo personalizado.
- Las carpetas se pueden contraer. Esta acción no es precedera. Las carpetas se muestran desplegadas cada vez que se inicie la ventana Power Query.
- Las carpetas no representan ningún tipo de jerarquía ni componente para el modelo de datos. No se cargan al modelo y no favorecen ni perjudican la optimización de consultas.

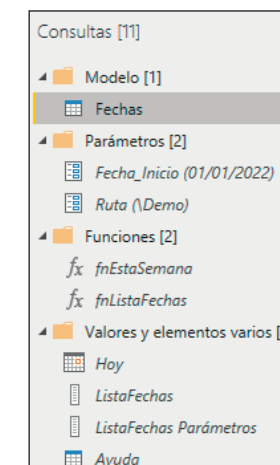


Figura 2.8. Ejemplo de panel de navegación con algunos grupos y objetos de distintos tipos.

El modelo tabular en Power Pivot es una base de datos que ofrece a las tablas y gráficos dinámicos de Excel la opción de acceso rápido a los datos y cálculos que se almacenan. La incorporación de este modelo aportó muchas ventajas y novedades al tratamiento de datos con Excel; entre ellas, se pueden destacar:

- La posibilidad de almacenar y comprimir los datos que tiene el modelo tabular en Excel garantiza que se puedan explorar conjuntos de datos, que por ser demasiado grandes no caben en una hoja de cálculo.
- La potencia de combinar distintos orígenes de datos externos.
- La capacidad de gestionar los cálculos y las consultas al modelo tabular con el lenguaje de expresiones DAX, que incluye la creación de nuevos tipos de cálculos avanzados y flexibles.

Más adelante, se incorporaron otros complementos a Excel como herramienta de inteligencia de negocio.

- Power Query para la fase de integración y transformación de datos.
- Power View para obtener informes con un estilo diferente.

La experiencia fue muy buena y se decidió combinar todo en una herramienta independiente de Excel llamada Power BI.

- El trabajo combinado de Power Query y Power Pivot en Excel resulta relevante para muchas organizaciones que no dan el paso a Power BI.

Para acceder al modelo tabular en Excel, se debe seguir la ruta **Power Pivot>Modelo de datos>Administrar**.

La figura 3.1 muestra la Vista de diagrama de la ventana Power Pivot, que aloja al modelo tabular en Excel.

El entorno Power Pivot tiene una banda de menú superior con un diseño muy amigable y que permite configurar y enriquecer cada elemento del modelo actual. Consta de dos vistas:

- **Vista de datos:** Expone la estructura y el contenido de las tablas y columnas del modelo. Los elementos tienen asociados un menú contextual que facilita las tareas de manipulación y transformación de los elementos que lo componen.
 - El panel **Área de cálculo** se asocia a esta vista y es un espacio reservado para el trabajo con medidas que enriquecen el modelo. Las medidas se escriben utilizando el lenguaje de expresiones DAX.
- **Vista de diagrama:** Permite explorar el contenido del modelo en su conjunto. Se distinguen las tablas y relaciones, admite el trabajo con jerarquías y dispone

de un menú contextual para cada elemento con acceso directo a opciones de modelado y diseño. Esta vista es la que se muestra en la figura 3.1.

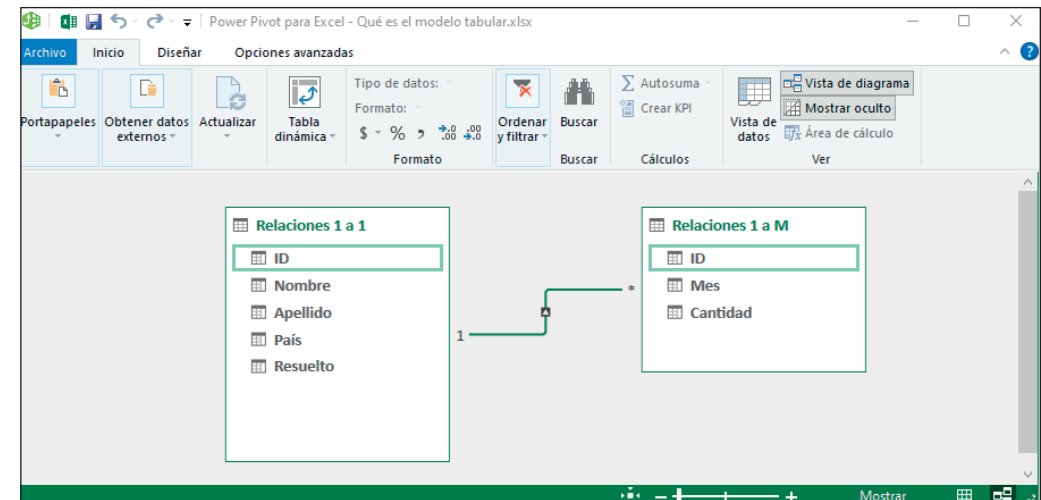


Figura 3.1. Vista de diagrama del modelo tabular en Power Pivot de Microsoft Excel.

El modelo tabular de Power Pivot es la piedra angular que permite almacenar y consultar grandes volúmenes de datos de forma eficiente en Microsoft Excel. Los datos llegan a Power Pivot desde los orígenes externos o, lo que es mejor, desde las conexiones y transformaciones creadas con Power Query en Excel.

Modelo tabular en Power BI Desktop

Power BI vio la luz en 2015 como herramienta de inteligencia de negocio. Desde siempre cuenta con una base de datos Analysis Services que alberga el modelo tabular.

Con Power BI Desktop se puede examinar la estructura del modelo tabular desde la vista **Modelo**. En la figura 3.2, se distinguen los componentes que integran el modelo tabular en Power BI, que son los mismos que en Excel, las tablas y las relaciones entre ellas:

- Las tablas se componen de columnas de tipos de datos específicos. Se almacenan únicamente los valores distintos en cada columna.
- Las relaciones tienen propiedades específicas como el sentido y la cardinalidad en cada extremo.

- Esta acción no involucra ningún componente del modelo de datos, solo las consultas.
- Si hay parámetro o funciones involucrados en las consultas, se van a pegar y a interpretar bien por la otra herramienta.
- Lo importante es garantizar tener acceso a los datos de origen.
- Si se trata de una consulta en Power BI que conecta a un origen no habilitado en Excel, se produce un error de conexión porque en Power Query para Excel hay menos conectores habilitados que en Power Query de Power BI.

Importar desde Excel a Power BI

Para importar desde Excel a Power BI, debemos seguir la ruta Archivo>Importar>Power Query, Power Pivot, Power View y elegir el documento Excel y el botón Iniciar. Comienza el proceso de importación de consultas y el modelo.

La figura 4.6 muestra la ventana informativa con los botones disponibles.

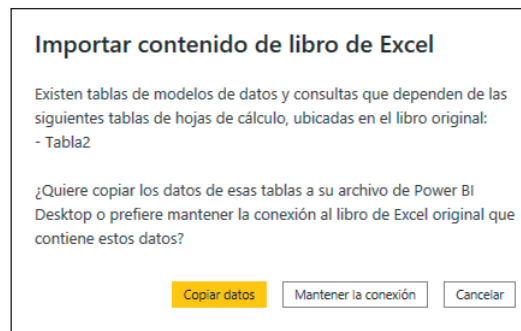


Figura 4.6. Ventana con opciones para importar contenido de Excel.

- Copiar datos: Migra todos los datos y crea directamente la tabla de contenidos. Lo convierte en un JSON y por eso es más complejo cualquier ajuste. Como la tabla con JSON tiene capacidad limitada, esta opción no es muy recomendable, es muy fácil y cómodo conectar directamente de un archivo Excel. La figura 4.7 muestra la consulta resultante.
- Mantener la conexión: Se crea una conexión como cuando se localiza el archivo y conecta, como se puede apreciar en la figura 4.8. Esta vez se trae también el modelo de datos.

En la figura 4.9 se muestra el resultado de la migración.

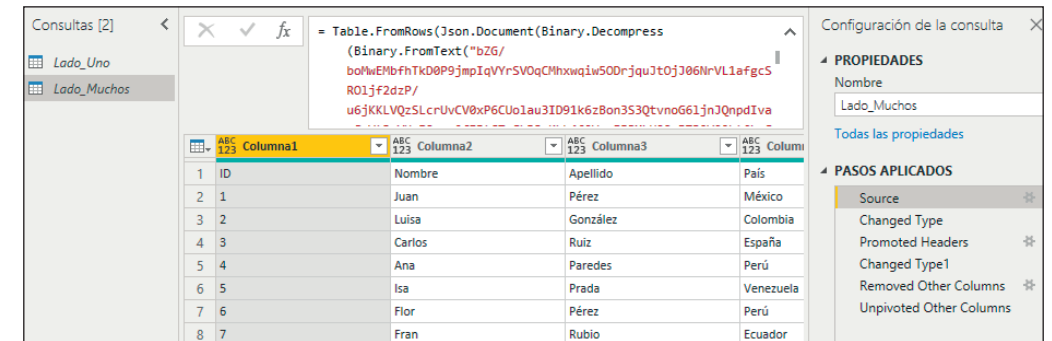


Figura 4.7. Consulta resultante al importar y copiar datos.

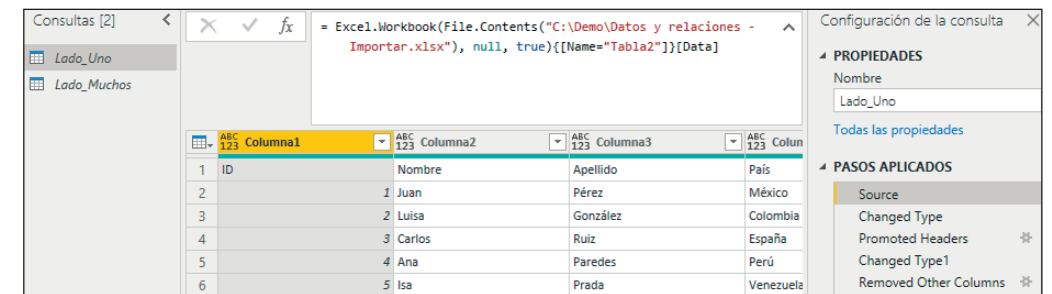


Figura 4.8. Consulta resultante al importar y mantener la conexión.

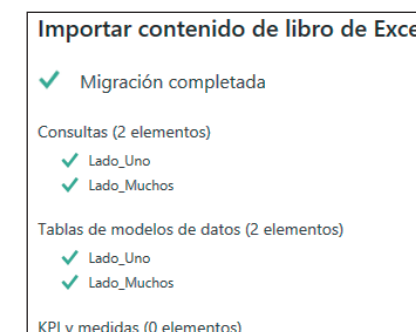


Figura 4.9. Resultado de la migración.

Lo que no está disponible es importar el modelo de Power BI a Excel.

- Como regla general, la herramienta Power BI cambia cada mes, el modelo es más avanzado e incluye nuevas funcionalidades.

6

Obtener datos externos de web

En este capítulo aprenderá:

- Opciones de conexión a orígenes externos vía web.
- Importar datos desde un origen web.
- Configurar tablas a partir de ejemplos.
- Retos por enfrentar cuando el origen de datos es web.

Es de gran utilidad hoy en día importar contenidos que se encuentren alojados en páginas web. Las fuentes de datos públicos y gratuitos que se ofrecen desde las administraciones locales y gubernamentales son muy valiosas para complementar estudios y ofrecer contexto al análisis de datos.

Formas de acceso a datos en orígenes externos vía web

Se accede a los datos que se encuentren alojados en páginas web generalmente de dos formas:

- **Acceso al contenido de una página web, creada en formato HTML:**
 - HTML representa las siglas en inglés de *Hypertext Markup Language*. Se refiere a un lenguaje de marcado para la elaboración de páginas web. Se estructura por etiquetas que envuelven las propiedades y contenidos de los elementos que aparecen en la página como texto, imágenes, vídeos, etc.
- **Acceso a un archivo alojado en un sitio, listo para descarga en un formato determinado (TXT, CSV, XLSX, JSON, XML, Parquet y otros muchos):** Se accede vía web y el navegador para seleccionar el o los elementos a importar dependerá del tipo de archivo.

Los contenidos que se encuentran originalmente en la web y se descargan en algún repositorio o estructura de carpetas local o remota se importarán con el conector correspondiente al tipo de archivo y nada tendrá que ver con un acceso web.

Retos al consumir datos de la web

Estructura del dato

Power Query está en constante evolución y mejora; sin embargo, el trabajo con orígenes web sigue siendo un desafío. Llegar hasta el detalle que se necesite explorando un documento HTML es posible. A veces, es muy fácil porque están creadas las tablas con su formato para ser leídas, y otras muchas veces es muy complicado.

Autoría de contenidos

Una de las mayores dificultades al conectar con orígenes web es que no hay forma de tener el control sobre el formato y la estructura de la página. Es posible pensar que es lo mismo que ocurre con todos los orígenes de datos externos, pero, en la

Azure

Power Query puede acceder a datos alojados en bases de datos en Azure, ejecutando los mismos pasos que para conectar a una base de datos relacional local o remota.

- La ventaja de utilizar bases de datos Azure es que se aprovecha la plataforma en la nube creada por Microsoft que proporciona software como servicio (SaaS), plataforma como servicio (PaaS) e infraestructura como servicio (IaaS) para construir, probar, desplegar y administrar aplicaciones y servicios mediante el uso de sus centros de datos.

El navegador que se muestra en la figura 7.5 expone el contenido de la base de datos para elegir las consultas a crear en Power Query.

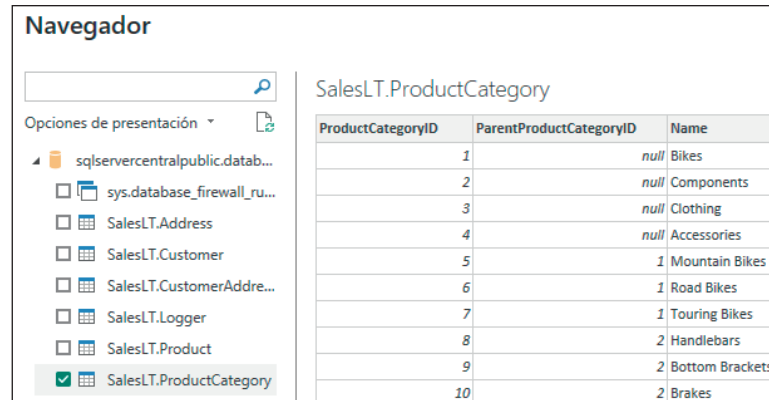


Figura 7.5. Navegador para base de datos en Azure.

Azure Analysis Services

SQL Server Analysis Services es un servicio integrado en SQL Server para la creación y administración de modelos analíticos, que pueden ser modelos multidimensionales o modelos tabulares. En las organizaciones se han desarrollado muchos modelos de ambos tipos, que son robustos, óptimos y necesarios para el análisis con Power BI.

Azure Analysis Services cumple con el mismo objetivo alojando estos modelos en la nube Azure de Microsoft. La diferencia es que, al trabajar en Azure, solo se pueden crear modelos tabulares que se pueden conectar a Power BI a través de Power Query y los conectores instalados. Es un origen de datos que, de existir, puede aportar muchos beneficios a la organización.

Se accede desde la ruta Obtener datos>Azure>Base de datos Azure Analysis Services:

- La ventana para conectar indica Base de datos SQL Server Analysis Services y pide los datos habituales para establecer la conexión.
 - Si la conexión se realiza la primera vez, hay que indicar el tipo de autenticación y las credenciales.
- La ventana Navegador no muestra diferencias con el navegador habitual de bases de datos de Analysis Services.
 - Si la conexión se realiza desde Power Query Online, las ventanas son algo diferentes, pero la esencia del proceso es la misma.

Microsoft Exchange

Power Query posee un conector nativo para obtener datos desde Microsoft Exchange que permite obtener datos desde un buzón de correo. Un buzón de correo contiene metadatos relativos al cuerpo del correo y los elementos adjuntos. Se almacenan, además, datos de calendario y de las personas involucradas en las tareas de colaboración que se comunican a través del correo. Esta información y el contenido de los adjuntos puede ser de gran utilidad en la creación de modelos analíticos.

La fuerza de Power Query permite realizar de forma dinámica y controlada todas las tareas de acceso, transformación y descubrimiento de contenidos. No se necesitan acciones manuales previas.

Se accede desde el menú Power Query con la ruta Nuevo origen>Servicios en línea>Microsoft Exchange Online.

- Hay que conectar a una cuenta de correo de Microsoft que utilice Microsoft Exchange.
- El conector permite elegir entre una cuenta de Exchange y una de Microsoft, y serán necesarias las credenciales de acceso. En mi caso, al trabajar con Microsoft 365, elijo Microsoft.

Muestra el navegador con las secciones que contiene Microsoft Exchange, que son: Calendar, Mail, Meeting Requests, People y Tasks.

- Al elegir Mail, muestra la vista previa de los metadatos de los correos almacenados en Microsoft Exchange Online.

Además, en dependencia del tipo de columna original y desde el menú **Agregar columna**, se pueden agregar tantas columnas como sean necesarias para que enriquezcan el modelo.

- Las opciones disponibles están separadas en grupos según el tipo de columna: Texto, Número y Fecha y hora.

Al combinar todos estos elementos, será posible crear las columnas que sean necesarias para dar respuestas a las necesidades del análisis.

Diferenciar casos para transformar o agregar columna

Es importante reconocer la diferencia en el modo de trabajar desde el menú **Transformar** y el menú **Agregar columna**:

- **Transformar**: Hay muchas opciones que permiten transformar una columna para modificar su estructura o contenido.
 - El resultado se aprecia en la misma columna.
- **Agregar columna**: Se trata de crear una columna nueva aplicando un paso sobre una columna existente, pero sin modificarla.
 - El resultado son dos columnas diferentes: la columna original, que no se modifica, y la columna resultante, que puede ser muy eficiente porque puede representar un aspecto nuevo del análisis, resumido, enriquecido, potenciado.

Un caso que puede ilustrar muy bien la diferencia es el uso de la acción **Combinar columnas**:

- Si se eligen varias columnas y se combinan desde la ruta **Transformar>Columna de texto>Combinar columnas**, desaparecen las columnas originales y se obtiene una columna resultante que contiene los textos combinados.
 - Se ejecuta la función `Table.CombineColumns`.
- Si se eligen varias columnas y se sigue la ruta **Agregar columna>De texto>-Combinar**, las columnas originales permanecen y se agrega una columna con los textos combinados.
 - Se ejecuta la función `Table.AddColumn`.

Ante cada paso que sea ejecutado desde una consulta que afecte o involucre los datos de una columna, hay que plantearse la necesidad de permanencia de esa columna en el modelo de datos:

- Si se trabaja desde **Agregar columna** y resulta que la columna original sobra, habrá que hacer un paso adicional para eliminarla.
- Si se trabaja desde **Transformar**, en un solo paso, se crea el contenido que se necesita y se descarta lo que no va a ser utilizado.

Evaluar la composición, calidad y distribución de los valores en una columna

Cuando se trabaja con informes analíticos es preferible contar con columnas de pocos valores que actúen como filtros, cabeceras de columnas y filas, así como ejes de gráficos. En la fase de preparación y transformación de datos, hay que elegir qué segmentos son los más eficientes para el proceso.

En la figura 14.1 se puede distinguir la distribución de la columna **Hijos** en la consulta **Cientes**.

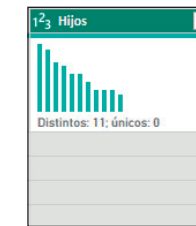


Figura 14.1. Muestra de los datos y perfil de distribución de la columna **Hijos**.

Al explorar la columna **Hijos** se puede concluir que:

- La distribución tiene buen aspecto:
 - Hay pocos valores distintos y están bien distribuidos.
 - No hay valores únicos.
 - No se aprecia el fenómeno de cola larga.
- La columna es válida para el análisis tal y como se ha recibido.

Llega el momento de comprobar los requisitos del análisis. ¿Qué ocurre si...?:

- Solo se necesita saber el tipo de familia y no la cantidad exacta de hijos.
- Se necesita crear una capa extra de análisis definiendo el tipo de familia y conservar la cantidad de hijos.

Si este fuera el caso, hay que continuar realizando tareas en la etapa de preparación de datos con **Power Query**.

16

El modelo tabular y el rol de Power Query

En este capítulo aprenderá a:

- Detectar escenarios de modelado que afectan al tratamiento de datos con Power Query.
- Definir correctamente las tablas y columnas del modelo como entidades y atributos.
- Entender las consecuencias de crear las tablas de fechas y horas automáticas.

En este capítulo se desglosan un grupo de aspectos relativos al diseño de modelos analíticos y cómo cada escenario tiene relación con las acciones que se realizan en Power Query.

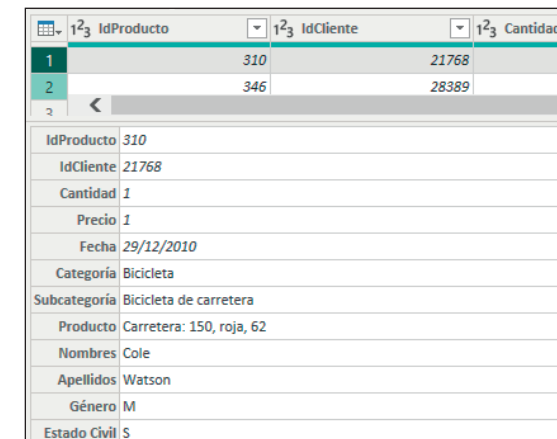
Los ejemplos se han simplificado con el objetivo de mostrar un problema en cada caso, aunque en los proyectos reales pueden presentarse combinados y su nivel de complejidad es considerablemente mayor.

Pasar de tabla única a modelo en estrella

El escenario de tabla única, como la que se aprecia en la figura 16.1, donde todos los elementos se concentran en una consulta, es perjudicial para el almacenamiento, rendimiento y calidad del modelo.

En estos casos pueden existir columnas que no aportan al análisis, porque están vacías o casi vacías, almacenan valores constantes o totales innecesarios.

- Desde Inicio>Administrar columnas>Elegir columnas, se puede decidir qué columnas van a permanecer en el modelo y en qué entidad se van a integrar.
- En el ejemplo, se identifican datos de clientes, productos y los datos de las ventas.



	IdProducto	IdCliente	Cantidad
1	310	21768	
2	346	28389	

IdProducto	310
IdCliente	21768
Cantidad	1
Precio	1
Fecha	29/12/2010
Categoría	Bicicleta
Subcategoría	Bicicleta de carretera
Producto	Carretera: 150, roja, 62
Nombres	Cole
Apellidos	Watson
Género	M
Estado Civil	S

Figura 16.1. Ejemplo de tabla única con datos que corresponden a varias entidades.

El proceso de creación de las tres entidades a partir de la tabla única es:

- Referenciar la consulta Tabla única, que es la consulta original que ya está limpia.
 - Se dice que está limpia cuando todas las columnas son útiles al modelo y su formato es correcto.

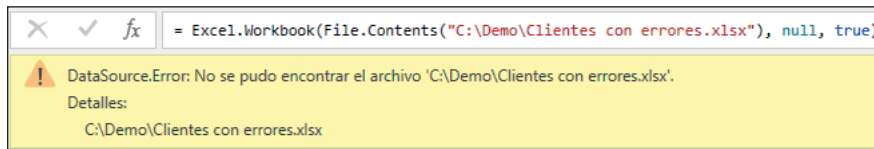


Figura 18.1. Error de paso por no poder establecer la conexión con el origen.

Botón Ir a Error

A veces, en la ventana de error aparece el botón Ir a error. Esto no ocurre siempre, solo en los casos en los que el paso actual no es responsable del error, sino que lo hereda, como se ve en la figura 18.2.

En este caso, el paso Tipo cambiado no se puede ejecutar porque hay un error de conexión a datos externos, es un error que se hereda desde el paso Origen.

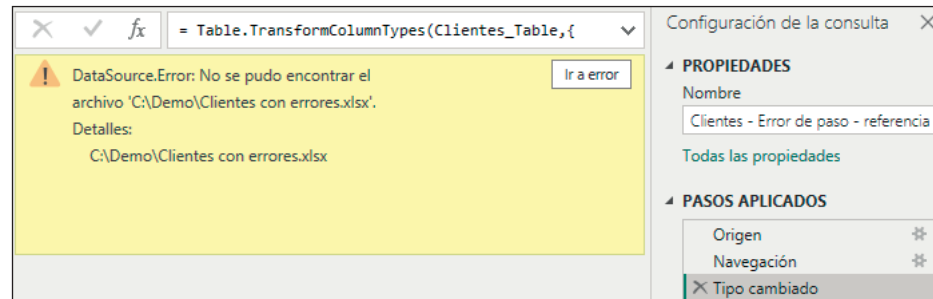


Figura 18.2. Ventana con botón Ir a error activo.

Al seleccionar Ir a error la consulta pasa al paso Origen y se presenta la ventana error con el botón Editar configuración.

Editar configuración para arreglar error de paso

El botón Editar configuración se puede activar desde la ventana de error cuando corresponda ajustar los valores de las propiedades de conexión.

Se abre el formulario que se muestra en la figura 18.3, lo que resulta muy conveniente para restablecer las propiedades de conexión.

Casos frecuentes de error de paso

Es bastante frecuente también el error de paso por referencia interna, ocurre cuando se pasa como parámetro del paso y hace referencia a una columna que no existe.

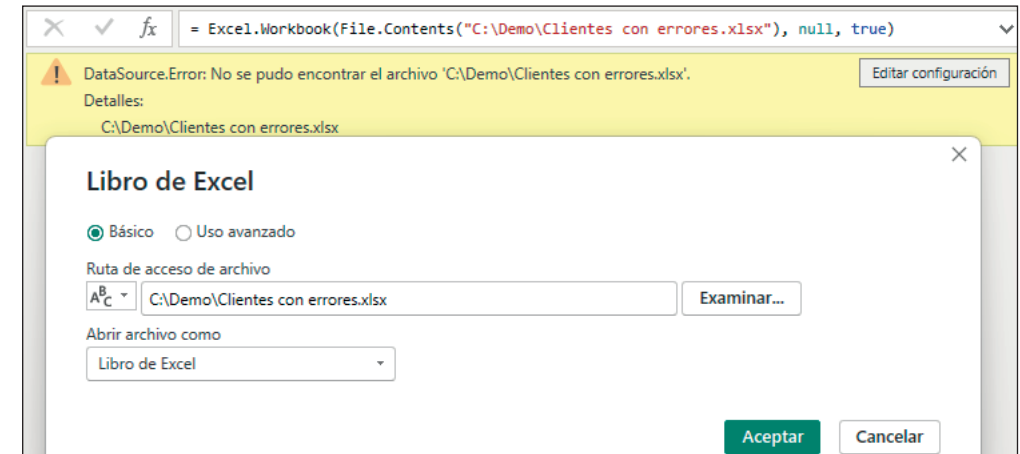


Figura 18.3. Editar configuración en la conexión a un archivo.

La siguiente expresión provoca error de paso si la columna Nacimiento no existe:

```
= Table.RenameColumns("#Otras columnas quitadas",{{"Nacimiento", "Fecha de Nacimiento"}})
```

- Si hay un error por referencia, porque no se encuentra una columna, el motivo es `Expression.Error`.

La solución ante este tipo de errores depende del escenario y los requisitos; si se quiere eliminar la columna, es posible que no haya que renombrarla porque no pasará al modelo.

- Cuando se mueven de posición los pasos para evitar errores, hay que verificar. Hay transformaciones como `Elegir columnas` que se configuran solo si se abre el formulario.
- Cualquier nueva columna que se agregue quedará desmarcada.

Un error como el de la figura 18.1 ocurre cuando el paso intenta conectar a un archivo que no existe, puede ser error en el nombre del archivo, en la ruta de carpetas, en el elemento al que se accede en la opción navegación o a una columna inexistente a la que se intenta definir el tipo de datos.

Error a nivel de celda o fila

El error ocurre a nivel de celda y por lo general se asocia a que la celda guarda un dato que no se corresponde con el tipo de datos asignado a la columna. Uno o más valores de la columna no se pudieron convertir al tipo de datos deseado.

20

Introducción a lenguaje M

En este capítulo aprenderá:

- Las bases del lenguaje M.
- Qué son las funciones y cómo crear funciones personalizadas.
- Cómo crear y distinguir los objetos lista, registro y tabla como componentes del lenguaje.
- Ejemplos de casos solucionados con código escrito en lenguaje M.

El lenguaje M merece un libro, una explicación extensa y detallada. Este lenguaje merece que se comenten sus funciones y componentes con rigor y detenimiento. El capítulo solo pretende mencionar algunos aspectos relativos al lenguaje y dejar varios ejemplos de uso que acompañen a los ejemplos que se han visto a lo largo del libro.

M es el lenguaje que mueve los hilos por detrás de lo que se ve en Power Query. Su nombre completo, que no se suele utilizar, es **lenguaje de fórmulas de Power Query** (*Power Query Formula Language*).

- Uno de los significados que más se atribuyen a M es *Mashup*, que significa una mezcla o fusión de elementos diversos. *Data mashup* se suele traducir por combinación o mezcla de datos que, junto a la conexión y transformación, es a lo que se dedica Power Query.

El lenguaje M es una estructura paso a paso, cada línea de código podría corresponder a cada paso en la transformación de la consulta.

- Esto puede variar según se establezca un formato en el Editor avanzado, para comprender mejor la expresión en determinados pasos.
- El lenguaje M distingue entre mayúsculas y minúsculas y utiliza nombres de funciones bastante descriptivos.

En el libro se han ido comentando las funciones del lenguaje y fragmentos de código que se han utilizado para dar cumplimiento a los desafíos que se han presentado.

- Cada elemento, con independencia de su tipo, se crea en lenguaje M. La expresión se puede ver en la barra de tareas o en el Editor avanzado.

El lenguaje M está compuesto por funciones asociadas a tipos de objetos, especialmente objetos de tipo tabla que es el más frecuente. Las funciones se pueden asociar también a registros, listas y a tipos de objetos primitivos como Texto y Fecha.

Otro grupo importante de funciones son las que se dedican a configurar e invocar los conectores a orígenes externos.

- Algunos ejemplos son: `Access.Database`, `Excel.Workbook`, `Csv.Document`, `PowerBI.Dataflows`, `PowerBI.Datamarts`, `DataLake.Contents`, `AmazonRedshift.Database`, `Python.Execute` y `ApacheSpark.Tables`.

El lenguaje incluye un conjunto de operadores que sirven en tareas de combinación y comparación. Además, existen operadores lógicos, de número, texto, fecha, duración y otros.

Conectar a datos externos desde flujos de datos

Las puertas de enlace en los flujos de datos

Los flujos de datos que se conectan a datos locales necesitan una puerta de enlace. Una puerta de enlace de datos local actúa como un puente que garantiza la transferencia de datos rápida y segura entre los datos locales, que no se encuentran en la nube, y varios servicios en la nube de Microsoft.

Una diferencia entre los entornos locales y la nube es el momento en que se necesita tener definida la puerta de enlace.

- Al trabajar en Power Query desde Power BI Desktop o Excel, en un entorno local, se puede desarrollar todo el proceso para crear la consulta y preocuparse por la puerta de enlace solo cuando se publique el conjunto de datos en la nube.
- Al trabajar con flujos de datos, la puerta de enlace tiene que haberse creado y permanecer activa, la sesión iniciada debe tener acceso a los datos a través de la puerta; de lo contrario, no es posible iniciar el proceso de creación de la consulta.

Sobre la elección y configuración de la puerta de enlace

Hay tres tipos de puertas de enlace: local, local personal y virtual.

- La puerta de enlace local permite que varios usuarios se conecten a varios orígenes de datos locales.
- La puerta local personal está diseñada para que una persona que crea informes acceda a sus datos. No se comparte el origen de datos con otros usuarios.
- La puerta de enlace virtual permite a varios usuarios conectarse a varios orígenes de datos protegidos por redes virtuales. No se necesita instalación porque es un servicio administrado de Microsoft.

No es objetivo de este libro tratar las tareas de configuración y administración en el Servicio Power BI, por lo que se verá muy brevemente este punto. Para configurar la puerta de enlace, siga estos pasos:

- Definir cuál de los tipos se va a necesitar.
- Descargar el archivo instalador desde la ventana de inicio de sesión en el Servicio Power BI y ejecutarlo para instalarlo.

- Si se despliega el botón **Descargar**, arriba a la derecha, en el Servicio Power BI, se puede descargar el instalador de la puerta de enlace de datos.
- Luego, instale la puerta de enlace según los pasos que se indiquen y las necesidades del escenario.

Vías para conectar a orígenes externos

El proceso de conexión es similar a Power Query en Excel y Power BI, hay que elegir:

- Tipo de conector:
 - Hay muchos conectores integrados y van en aumento.
- La ruta de conexión.
- Navegación a un componente:
 - Tabla o vista de base de datos o elemento contenido en un archivo.

Arrastrar un archivo

Una opción muy interesante es que se puede arrastrar un archivo directamente desde una carpeta local al Servicio Power BI.

Power Query reconoce el tipo de archivo y activa el tipo de conector adecuado y sus propiedades, como se puede ver en la figura 22.3.

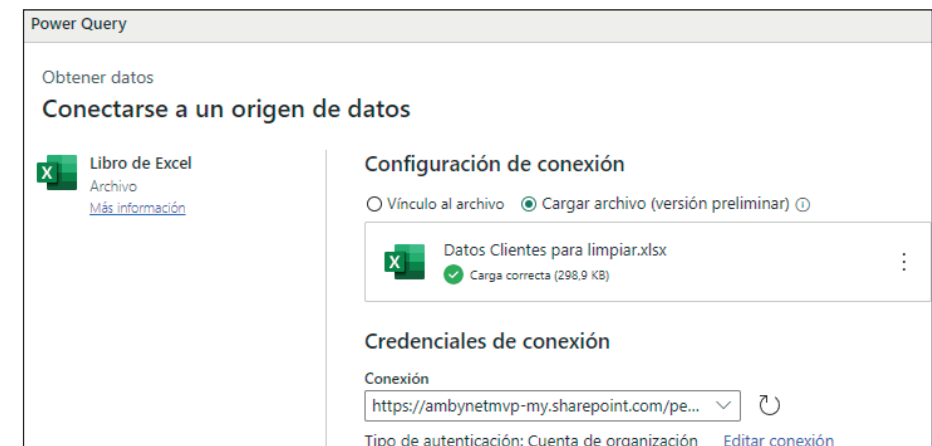


Figura 22.3. Propiedades de conexión para un archivo arrastrado directamente al Servicio Power BI.

Tratamiento de datos con Power Query



Manual Imprescindible

Tratamiento de datos con Power Query comienza introduciendo el entorno de trabajo en Power Query y su capacidad para conectar a diferentes orígenes externos. A continuación, descubre un grupo de tareas, opciones y casos de uso en la etapa de exploración y transformación de columnas y las consultas.

Un bloque importante del libro se dedica a exponer conceptos del modelo tabular. Con ejemplos prácticos, propone soluciones a escenarios de modelado de relaciones, definición de entidades y atributos, trabajo con parámetros y solución de errores, entre otros.

Contiene una introducción al lenguaje de consultas M, que actúa como motor de Power Query y desvela algunas ideas sobre cómo solucionar desafíos en la etapa de transformación de datos a partir del uso del lenguaje M.

Se descubren buenas prácticas y el uso de herramientas de diagnóstico y medición para conseguir la optimización de consultas con Power Query y la carga de datos más eficiente.

Aunque se ha escrito desde la herramienta Power BI Desktop, que es la más accesible, aporta un resumen de otros sabores de Power Query en Excel y los entornos analíticos en la nube de Microsoft como los Flujos de datos en Power BI y Power Platform y finalmente, los Flujos de datos Gen2 en Data Factory integrado en Microsoft Fabric.