Income Statement Semantic Models

Building Enterprise-Grade Income Statement Models with Power BI

Chris Barber

Apress®

Income Statement Semantic Models

Building Enterprise-Grade Income Statement Models with Power BI

Chris Barber

Apress®

Income Statement Semantic Models: Building Enterprise-Grade Income Statement Models with Power BI

Chris Barber
KINGSTON UPON THAMES, UK

ISBN-13 (pbk): 979-8-8688-0329-1 ISBN-13 (electronic): 979-8-8688-0330-7
https://doi.org/10.1007/979-8-8688-0330-7

Managing Director, Apress Media LLC: Welmoed Spahr
Acquisitions Editor: Ryan Byrnes
Development Editor: Laura Berendson
Coordinating Editor: Gryffin Winkler

Cover designed by eStudioCalamar
Cover image by adobe stock photos

Distributed to the book trade worldwide by Apress Media, LLC, 1 New York Plaza, New York, NY 10004, U.S.A. Phone 1-800-SPRINGER, fax (201) 348-4505, e-mail orders-ny@springer-sbm.com, or visit www.springeronline.com. Apress Media, LLC is a California LLC and the sole member (owner) is Springer Science + Business Media Finance Inc (SSBM Finance Inc). SSBM Finance Inc is a **Delaware** corporation.

For information on translations, please e-mail booktranslations@springernature.com; for reprint, paperback, or audio rights, please e-mail bookpermissions@springernature.com.

Apress titles may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Print and eBook Bulk Sales web page at http://www.apress.com/bulk-sales.

Any source code or other supplementary material referenced by the author in this book is available to readers on GitHub (https://github.com/Apress). For more detailed information, please visit https://www.apress.com/gp/services/source-code.

If disposing of this product, please recycle the paper

For Gemma and Sophie

Table of Contents

About the Author

Chris Barber is a chartered accountant (ACMA, CGMA) and Microsoft MVP. He has trained over 1,000 people on how to build income statements in Power BI, delivered several public talks on using the Microsoft BI stack within finance, and runs StarSchema.co.uk.

All author proceeds from this book are being donated to Save the Children.

About the Technical Reviewer

Triparna Ray is a passionate technologist with over two decades of experience in Microsoft Business Intelligence.

She is a well-known Power BI consultant in Microsoft Community specializing in Financial Reporting.

Triparna has worked as an architect in a number of Power BI implementations across domains and geography.

She is a mentor, trainer, and public speaker and has presented in various technical conferences.

Acknowledgments

I have so many people to thank for their contributions that it is impossible to include everyone. So, thank you to everyone who has played a part, even if you did not know you were doing so. The data community has been invaluable with many excellent conferences, user groups, blogs, and other forms of learning. Any presenters, attendees, and those I have spoken to at an event or engaged with through other means, you have all made a significant contribution to the contents of this book.

There are, however, some special mentions I would like to make.

Firstly, to Prathyusha (Prathy) Kamasani who provided the most welcoming and open environment in which to learn and ask questions. Without this, I would not have pursued a change in direction within my career, and this book would never have been written.

I also want to thank Mark Wilcock who has been incredibly supportive. Mark accelerated my involvement in the data community, and without him I would not have received my first Microsoft MVP award.

There are many colleagues and former colleagues I must also thank who provided support or took time to explain the overwhelming process involved in publishing a book. Special mentions to James Dales, Pragati Jain, Rishi Sapra, Mihaly Kavasi, Richard Swinbank, and Lee Englestone. Your help is greatly appreciated.

Finally, a special mention to Triparna Ray (technical reviewer), Ryan Byrnes (acquisitions editor), Laura Berendson (development editor), Gryffin Winkler (coordinating editor), Nirmal Selvaraj (production editor), and all those involved in the publication process. I greatly appreciate all the hours you have spent reviewing this book and working through the various examples.

Preface

For years, I have seen various entities struggle to produce income statement reports that meet modern analytical demands. A quick internet search and you will see a wide variety of methodologies from reputable sources, some which are heavily model focused, others which create a plethora of calculations, and others which use logic within the visual itself to perform calculations. This range of options runs in conflict with what one might intuitively expect if you come from an accounting background; there is a standardized process of building income statements based on the trial balance, and outputs are homogenous because they are regulated by accounting standards such as IAS (International Accounting Standards) 1 Presentation of Financial Statements. This is regardless of whether the company produces semiconductors or sells advertising revenue. Therefore, a valid question is why is there such a lack of consensus on the best way to model income statements?

Part of the reason for the lack of consensus is because the authoritative source on dimensional modelling (*The Data Warehouse Toolkit, Volume 3* by Kimball and Ross) said you should not try to model the entire income statement (I am paraphrasing, but this is the gist of it). While the reasons were conclusive at the time of writing, modern demands and the powerful analytical engines underpinning Microsoft Fabric, Power BI, and Analysis Services Tabular make it possible to produce income statements in their entirety. The, albeit lofty, ambition of this book is to drive consensus on how to achieve this with an approach based on the trial balance that can be reused across entities.

For a semantic model income statement to be widely accepted, this book posits two fundamental criteria. Firstly, the semantic model should be based on the trial balance; the trial balance is the precursory step in the accounting process prior to producing income statements and is thus common across entities. By building a solution based on the trial balance, a base semantic model can be built which can then be customized to the nuances of each particular entity, that is, the ways in which a particular entity explains its income statement. Secondly, the semantic model must adhere to the principles of dimensional modelling as outlined by Kimball and Ross; these principles are widely accepted, and any solution which is in violation is unlikely to be widely adopted within the technology community.

Whom This Book Is For

This book has two primary audiences:

1. Technical (i.e., solution architects, Microsoft Fabric developers, Power BI developers) who are struggling to produce income statement semantic models because of the modelling complexities and knowledge needed of the accounting process
2. Finance (i.e., management accountants) who have hit the limits of Excel and have started using Power BI, but are unsure how income statement semantic models are built

This book covers both finance and technical areas in sufficient depth, without covering areas which are irrelevant to modelling the income statement. For instance, a knowledge of complex group structures is required, but in modelling you do not need to know how to calculate non-controlling interest as this is an output provided by the finance team or system.

If you are coming from a technical background, you may be able to quickly cover the more technical chapters, being familiar with concepts such as role-playing dimensions, surrogate keys, and aggregations. In contrast, you may find you cover the accounting heavy chapters at a slower pace as these introduce concepts such as the qualitative characteristics of financial reports, double-entry bookkeeping, and the accounting process. The reverse is expected to be true of those coming from a finance background.

Assumptions About You

There is an assumption you are aware of Power BI basics including knowledge of Power Query, modelling, DAX (Data Analysis eXpressions), and visualization. This includes a basic understanding of DAX theory such as filter and row context, how to configure security, and amending properties using Power BI Desktop such as adding descriptions, creating folders, and hiding fields. Furthermore, there is an assumption you have a basic grasp of Excel including familiarity with pivot tables and the ability to write formulas.

What Is Not Covered

This book does not cover the data engineering required to go from source systems to semantic model inputs. There are a wide array of ways in which data can be made available for consumption and many excellent resources, such as books and blogs on the topic of data engineering. Some of these resources are technology agnostic and focus on the principles of data engineering, while others focus on specific technologies.

If the data engineering process stores information in delta-parquet, it can immediately be accessed by a semantic model using direct lake mode. However, this is not essential, and all examples in this book use import mode with comma-separated value (CSV) files and thus can be followed along using the free version of Power BI.

Organization of This Book

Parts

This book is divided into four parts, each of which has a specific learning objective:

Part 1, "Modelling and the Income Statement": Learn what modelling the income statement entails, why it is important, and how income statements are constructed.

Part 2, "Calculating Account Balances": Learn how to optimally calculate account balances using a Star Schema.

Part 3, "Producing External Income Statement Semantic Models": Learn how to produce external income statement semantic models which enable income statements to be analyzed from a range of perspectives and drilled into to reveal the underlying accounts and journal entries.

Part 4, "Producing Internal Income Statement Semantic Models": Learn how to create multiple income statement layouts and further contextualize financial information by including percentages and non-financial information. Also, learn about the various security and self-service considerations and how the semantic modelling approach overcomes various challenges in producing income statements.

Chapters

This book is designed to flow from introductory chapters to more advanced concepts. Each chapter assumes the content from the previous chapters has been understood. Once each chapter has been covered, this book is designed to be used as a refresher when building income statement solutions.

At a glance, the chapters are as follows:

Chapter 1, "What Is an Income Statement Semantic Model?," introduces modelling the income statement, covering the core components of statutory reporting, such as net income, controlling interests, non-controlling interests, and earnings per share. It extends the income statement to include internal requirements, such as adding percentages and including non-financial information to contextualize financial information. It also covers two ways of viewing an income statement semantic model, as a way of navigating financial information and a way of measuring the accounting process.

Chapter 2, "How the Income Statement Is Constructed," covers the accounting process which results in the production of the income statement to net income. This introduces key accounting concepts such as double-entry bookkeeping, the accrual method of accounting, the trial balance, and account balances. Also covered is group accounting which can result in net income being attributed to controlling and non-controlling interests. Finally, earnings per share calculations are covered.

Chapter 3, "Building a Reusable Solution," outlines the sixteen challenges in modelling the income statement, divided into nine calculation challenges, four presentational challenges, and three analytical challenges. It breaks down the semantic model into the three stages of data preparation, data modelling, and Data Analysis eXpressions (DAX). It brings the challenges and semantic model process together to highlight

how a reusable solution based on the trial balance can overcome fourteen of the challenges; overcoming the remaining two challenges requires customization from the base solution.

Chapter 4, "Why Model the Income Statement?," covers the benefits of modelling the income statement in its entirety, such as helping to explain the income statement, improve decision-making, improve data quality, gain a deeper understanding of profitability, and uncover insights to facilitate the month-end close process. This also addresses the core arguments against modelling the income statement in its entirety, that (1) business intelligence should focus on individual lines in the income statement rather than reproducing financial reports and (2) the visual – as opposed to the semantic model – should contain the income statement logic.

Chapter 5, "Conceptual Account Balance Models," explains how to calculate account balances using a trial balance Star Schema. It introduces the concept of granularity and increasing the dimensionality of the solution beyond the base trial balance model.

Chapter 6, "Logical Account Balance Models," explains how to convert conceptual models (high-level designs) into logical models (detailed blueprints). It covers the conversion of the trial balance and journal entry conceptual models, including introducing data types, keys, degenerate dimensions, attributes, and measures. Finally, it shows how aggregations can be used to optimize the account balance calculation.

Chapter 7, "The Trial Balance Semantic Model," builds a line-item solution in Power BI based upon trial balances for Tyrell Corp and Weyland Industries. It covers the core stages in building a semantic model of Power Query transformations, data modelling, and DAX. Finally, it shows how the semantic model can be used to re-create external income statements in Excel.

Chapter 8, "A Journal Entry Semantic Model," builds a line-item solution in Power BI based upon journal entries for StarSchema.co.uk. It covers the core stages in building a semantic model of Power Query transformations, data modelling, and DAX.

Chapter 9, "The Four Subtotal and Subset Types," explains how subtotals and subsets can contain a mix of revenue and expense accounts. Consequently, this leads to four drill-down options: net credit less debit, net debit less credit, all credit less debit, and all debit less credit. Examples of all four options are shown exhibiting their different behavior.

Chapter 10, "External Reporting Logical Models," expands the logical account balance models (Chapter 6) to produce logical models which reproduce external reports in their entirety. It introduces, with a worked example, the layout table which contains the logic surrounding the presentation of income statements: what lines appear, in which order they appear, which calculations are performed, and how they are formatted.

Chapter 11, "External Reporting Semantic Models," extends the trial balance semantic model (Chapter 7) and the journal entry semantic model (Chapter 8) to produce external income statements in their entirety. It covers the core stages in expanding the semantic models of Power Query transformations, data modelling, DAX, and dynamic format strings.

Chapter 12, "Internal Reporting Logical Models," expands the external reporting logical models (Chapter 10) adding additional layouts and supporting an additional fact table containing information from departments such as human resources and marketing.

Chapter 13, "Internal Reporting Semantic Models," extends the external reporting semantic models (Chapter 11) to contain multiple layouts and internal metrics. It covers the core stages in expanding the semantic models of Power Query transformations, data modelling, and DAX.

Chapter 14, "Security and Self-Service Considerations," covers various security considerations and ways to make it easier for end users to retrieve information from the semantic model.

Chapter 15, "Review of the 16 Challenges," reviews how the approach to building semantic models outlined in this book overcomes the nine calculation challenges, the four presentational challenges, and the three analytical challenges.

To Get the Most Out of This Book

You will need the latest version of Power BI Desktop downloaded and installed (Download Power BI Desktop from Official Microsoft Download Center). All examples used throughout this book have been tested in the January 2024 release of Power BI Desktop and will work on future versions. It is also advised to turn off default settings for inferring relationships and creating private date tables. At the time of writing, "dynamic format strings for measures" is a preview feature that must be tuned on.

In addition to Power BI Desktop, you will need the latest version of Tabular Editor (https://github.com/TabularEditor/TabularEditor).

In some chapters, you may need to have Excel and a Power BI Service account. You can sign up for a Power BI Service as an individual. Read more here: https://docs.microsoft.com/en-us/power-bi/fundamentals/service-self-service-signup-for-power-bi?WT.mc_id=5003466.

Turning Off Default Settings for Inferring Relationships and Private Date Tables and Enabling Preview Features

To build the solution from scratch, without Power BI inferring any relationships or creating any additional hidden tables in the background, the default settings for relationships and time intelligence are required to be turned off. At the time of writing, *dynamic format string for measures* is a preview feature that should be enabled. All these options can be set in four steps:

Step 1: In Power BI Desktop, select **File (1)**.

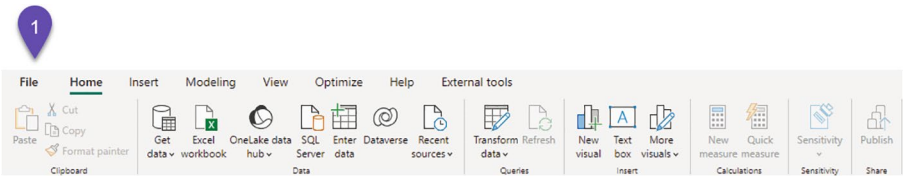

Figure 1. *Select file*

Step 2: Select **Options and settings (1)**. Select **Options (2)**.

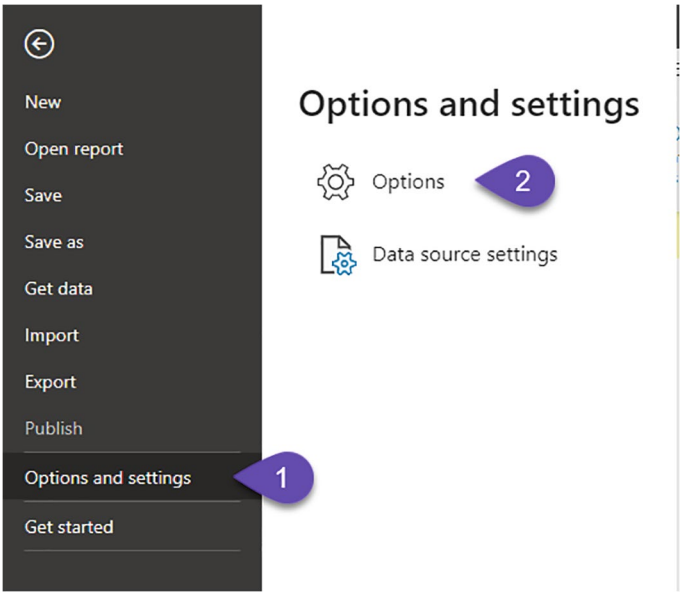

Figure 2. *Select options*

Step 3: In the pop-up box, under current file select **Data Load (1)**, then untick **Import relationships from data sources on first load (2)**, **Autodetect new relationships after data is loaded (3)**, and **Auto date/time (4)**.

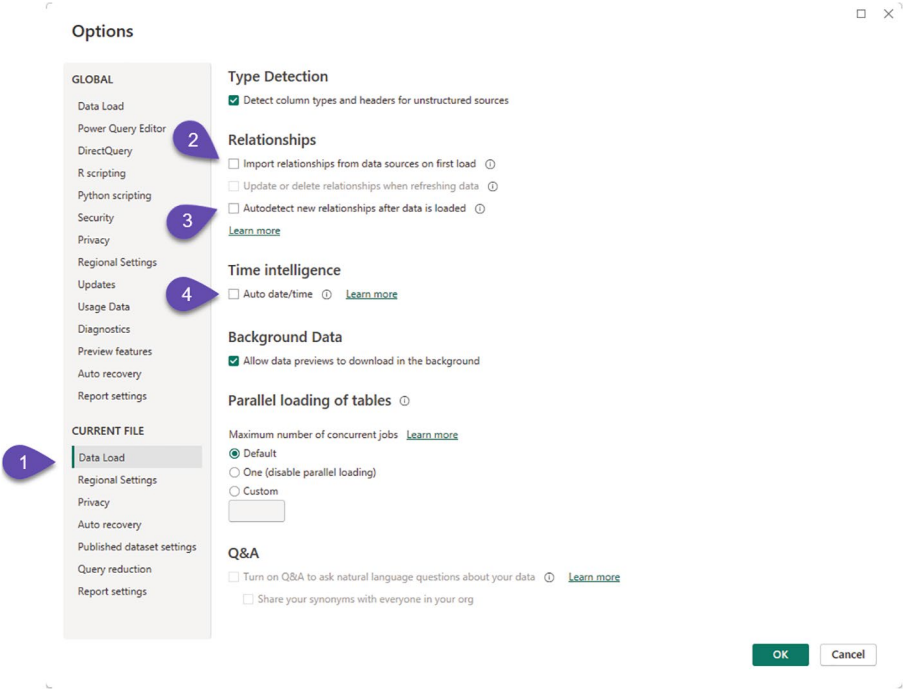

Figure 3. *Data load options*

Step 4: In the pop-up box, under global select **Preview features (1)**, then tick **Dynamic format string for measures (2)**.

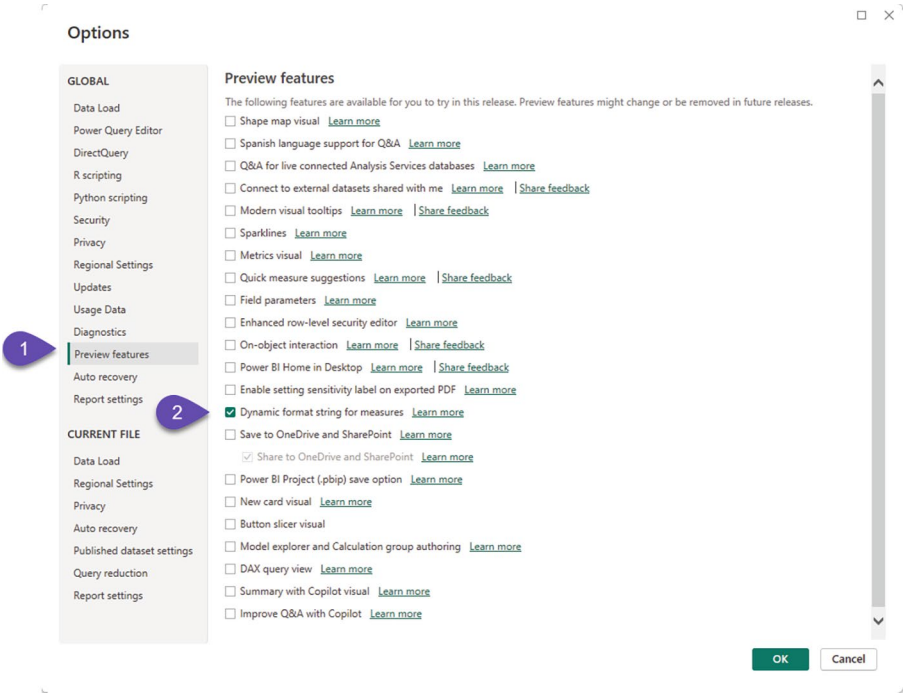

Figure 4. *Preview options*

Accessing Tabular Editor from Power BI Desktop

Power BI Desktop is catered toward basic to intermediate users from a modelling perspective. While the features available in Power BI Desktop are increasing, third-party tools are still required to access some of the more advanced features, and these tools also drastically speed up development.

To open Tabular Editor, click **External tools (1)**, then **Tabular Editor (2)**.

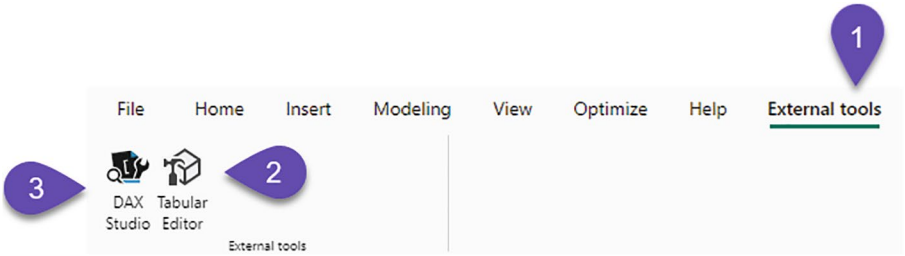

Figure 5. *Launch Tabular Editor 2*

Resources

You can download the accompanying resources from GitHub at `https://github.com/Apress/Income-Statement-Semantic-Models`. This includes all the CSV and PBIX files. To use the PBIX files, open Power Query, and in the folder **Parameter (1)**, select **Folder (2)**, then type the path name where it says **TYPE FILE LOCATION (3)**; this is the location CSV files have been downloaded to, that is, "C:\Users\Chris.Barber\OneDrive\Downloads\".

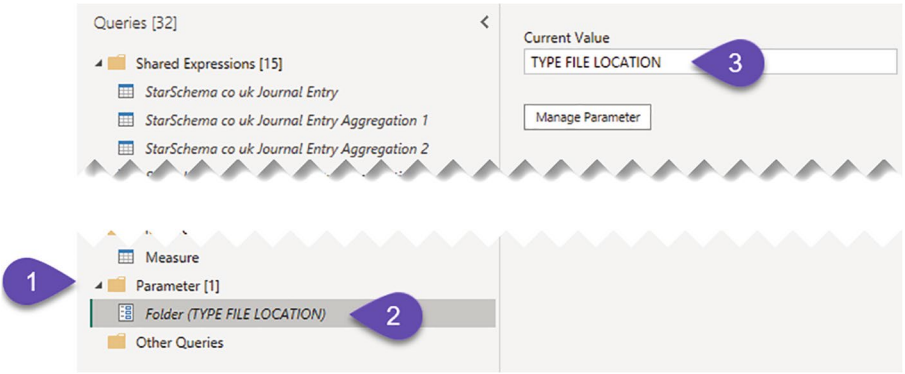

Figure 6. *Using parameters*

Examples Used

Given financial information is closely guarded, this book uses fictional examples when analyzing income statements down to accounts and individual journal entries. There are three examples used throughout this book:

1. Tyrell Corp
2. Weyland Industries
3. StarSchema.co.uk

The Tyrell Corp and Weyland Industries solutions go down to account level and are used to create trial balance semantic models in Chapter 7, whereas the StarSchema.co.uk solution goes down to individual journal entries in Chapter 8. Chapter 11 builds on any of the three semantic models, creating external reporting semantic models; these contain the logic surrounding the presentation of the income statement. Chapter 13 then covers adding a secondary income statement layout for each example which includes percentages and non-general ledger information, such as the number of full-time equivalent (FTE) employees.

Conventions Used

- Tables are identified using single quotations: 'Table', that is, 'Account'.
- Columns are identified using single quotations for the table and square brackets for the column: 'Table'[Column Name], that is, 'Account'[Account Name].
- Measures are identified using square brackets without a table name: [Measure], that is, [Actuals].

PART I

Modelling and the Income Statement

In this part, the concept of an income statement semantic model is introduced. There is an assumption you are aware of Power BI basics including knowledge of the layers Power Query, data modelling, DAX, and visualization.

In this part, you will learn what modelling the income statement entails, how income statements are constructed, the challenges a reusable solution must overcome, and why it is important to model the income statement in its entirety.

This part comprises the following chapters:

- Chapter 1, “What Is an Income Statement Semantic Model?”
- Chapter 2, “How the Income Statement Is Constructed”
- Chapter 3, “Building a Reusable Solution”
- Chapter 4, “Why Model the Income Statement?”