IN THIS CHAPTER

- Recognizing the business value of Microsoft Power Platform
- Identifying the key components of Power Platform

Chapter 1
Touring the Power Platform

Until recently, only skilled technicians could develop software, provide data analytics, automate workflows, and work with artificial intelligence. Now, business technology tools, such as the tools that make up the Microsoft Power Platform suite, can help users of just about any skill level perform application development and data analysis. With these tools, you can develop enterprise solutions by using simple drag-and-drop interfaces — the tools do all the coding magic behind the scenes.

This chapter provides an overview of the capabilities included in Microsoft Power Platform. I discuss the benefits that Microsoft Power Platform's highly adaptable and open platform offers. The platform greatly simplifies business operations and allows for significant integration opportunities, not just across Microsoft applications, but for a broad range of enterprise vendors outside the Microsoft ecosystem.

Grasping the Power of Power Platform

Microsoft Power Platform is a suite of Microsoft tools that enables users of all skill levels to rapidly develop applications. You don't need to be a programmer to use these tools. You simply need to know how all the tools work together and have a strong sense of logical reasoning that you can apply.

Now, you might be wondering what I mean by a *suite.* Well, Power Platform includes a collection of tools within a single application: tools for application development, enterprise data analytics, workflow automation, virtual agent design, data connection deployment, and data platform design. The Microsoft 365 suite also bundles a collection of tools together: Word for document management, Excel for data analysis, PowerPoint for presentations, and Access for consumer-oriented database management.

Table 1-1 lists each tool included in Microsoft Power Platform and briefly describes its purpose. The section "Zooming into the Platform Features," later in this chapter, discusses each of Power Platform's tools, or *components,* in more detail.

TABLE 1-1 **Microsoft Power Platform Tool Capabilities**

Tool	Purpose
Power Apps	Low-code development tool to create custom business applications. Makes app creation more accessible and less time-consuming.
Power BI	A sophisticated tool for data visualization and business analytics. Users can create comprehensive reports and dashboards with a variety of charts and visuals.
Power Automate	Robust business process automation tool for the desktop and cloud, with a focus on operational efficiency.
Power Pages	A web design tool that you can use to create professional websites or public-facing portals without having to use code. Integrates with Power Apps, Power Automate, and Power BI.
Dataverse	A cloud-based data repository that enables users to store and manage data used by business applications such as Power Platform and Dynamics 365.

Benefiting from a low-code, no-code solution

Low-code, no-code development means that a developer, either professional or amateur, can design and develop a product for the Web by using a set of intuitive drag-and-drop tools that reduce or even eliminate the need for code. Sure, you need to bring your objects together on the application or *report canvas,* the area in which you design the low-code capability, by referencing specific parameters (using short *references* or code snippets). But by no means do you have to write a novella's worth of code just so a user can click a button or activate a drop-down menu. Those days are long gone.

REMEMBER

Unlike your traditional coders, Power Platform developers fall into two types — citizen developers and professional developers:

- **Citizen developer:** Knows enough to be dangerous when it comes to web development and their business, but has no professional training in developing complete software applications.
- **Professional developer:** Has many years of experience with programming languages that are available in Microsoft Visual Studio and understands more than just drag-and-drop capabilities. Professional developers are familiar with more than one programming language (such as C# and PowerShell) and frameworks (such as .NET Framework and ASP.NET).

Low-code platforms

Low-code platforms, such as Power Platform, provide pre-built components, templates, and drag-and-drop tools so that developers can reduce the amount of hand-coding they need to input to develop applications. Notice that I say reduce — you still likely need to do some hand-coding, even when you work in a highly visual environment for creating your end product, not just a glorified code editor.

If you work in the visual development environment of a low-code platform, you can drag and drop application components, such as check boxes, drop-down menus, labels, or galleries; connect them by using predefined workflows, data connections, or custom formulas; and configure each component's properties via logical units. Most developers do find, however, that they have to do some hand-coding to incorporate complex functionality or logic. More complicated coding practices might include using multistep logic, such as if-else or do-while logic.

Low-code platforms can help you with rapid development when you need to scale over time. Whether you're a professional developer who has decades of experience or an industry professional who needs to quickly create functional solutions, these platforms can assist. For example, industry professionals commonly want to convert a legacy .NET application into a model-driven Power Apps app. The datastore for the .NET app often takes the form of an Access or SQL Server database, which is then migrated to Dataverse. Although this process may seem straightforward, it often requires careful execution to ensure flawless conversion.

No-code platforms

No-code platforms eliminate the need for you, as the developer, to do any hand-coding, period. You don't need programming skills to use these platforms. You

simply drag, drop, and click, using a visual interface. No-code platforms offer pre-built templates that offer a wide array of out-of-the-box interfaces to help users configure applications. If you've ever used Microsoft Word (who hasn't?), created a table, and then saved your document as an HTML, you've effectively created a no-code document — ta-da!

Unlike low-code platforms (see the preceding section), where you might need to have some development skills to use them — hence users may shy away from those tools — no-code platforms enable just about anyone to get involved in the application development, data analytics, and workflow automation lifecycle. You just need to have an idea, generally know where to place the content, and then click a Save button. Essentially, most users who leverage a canvas app (which you can read about in the section "Power Apps," later in this chapter) follow that process in conjunction with the help of Microsoft 365 applications such as Excel (Microsoft's spreadsheet program), PowerPoint (its presentation program), and SharePoint (an online collaboration and content management offering).

Connecting with the Microsoft ecosystem and beyond

Microsoft made sure that every application in its enterprise lineup works with Microsoft Power Platform, and you can use them all without having the skills of a lifelong coder.

And believe me, Microsoft isn't alone in its quest to move away from requiring users to have deep technical know-how. The industry is leaning overall toward a focus on allowing business users to be more efficient in their ability to develop applications, analyze data, and automate their business operations. In its marketing, Microsoft showcases hundreds of ways that businesses can effectively use Microsoft products, but here are three takeaways I think are worth pointing out:

- **Business application productivity:** With Power Platform, you can easily integrate with Microsoft 365 (formerly Office 365) applications such as Word (Microsoft's word processor), Excel (its spreadsheet software), PowerPoint (presentation software), Outlook (its e-mail service), and SharePoint (Microsoft's spot for online content storage) to create business workflows, analyze data created from defined lists, or create small applications from structured datasets.

  For example, you can go to the Automate menu in Excel to trigger Workflows or the Integrate menu in SharePoint to execute Power Platform functionality with one click. It's that easy.

- **Enterprise applications:** Dynamics 365 is Microsoft's suite of enterprise resource planning (ERP) and customer relationship management (CRM) applications. A developer can create customer forms for either the ERP or CRM application suite, establish business workflows when a user enters specific data into the system, or curate highly graphical reporting to augment data stored in the applications without much programmatic effort.
- **Cloud computing connectivity:** Power Platform utilizes Microsoft's cloud platform, Azure, for advanced features. For example, Azure Logic Apps supports the creation of advanced workflow automation, and Azure AI services enhance intelligent application capabilities, including integration with Microsoft Copilot (Microsoft's AI-powered digital assistant).

Microsoft offers one of the most comprehensive security, compliance, and governance solution sets. Applications built by using Power Platform benefit from Azure's robust security features, including those that are low-code and no-code (which I talk about in the section "Benefiting from a low-code, no-code solution," earlier in this chapter). Additionally, administrators of Power Platform applications can govern and monitor their applications by using a wide range of compliance and governance tools built right into the Microsoft 365 console.

Zooming into the Platform Features

Microsoft Power Platform, as illustrated in Figure 1-1, provides a unified application platform designed to streamline business processes, improve data visualization, and simplify application development within the Microsoft ecosystem. The top row displays each of the Power Platform applications. These applications can connect to other data sources in one of three ways:

- Connecting to Microsoft and third-party data connectors to push and pull data
- Using Microsoft's own Dataverse data repository
- Integrating with one or more Microsoft AI services, such as Copilot, to assist users in automating workflows, generating insights, and building applications more efficiently through natural language interactions

All of these Power Platform applications require access to data for them to work successfully.

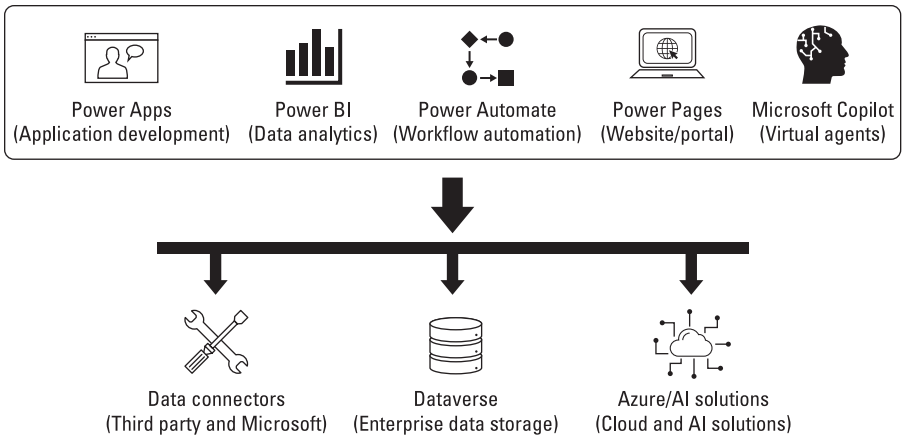

FIGURE 1-1: Microsoft Power Platform components and capabilities.

TECHNICAL STUFF

Microsoft has integrated some low-code programming languages (see the section "Low-code platforms," earlier in this chapter) into Power Platform:

- **Power Fx (Power Platform's low-code programming language):** Allows users to define logic and automate functionality across the platform by using simple, Excel-like formulas to manipulate data, trigger actions, and call parameters.
- **DAX (Data Analysis Expressions):** Used specifically in Power BI (discussed in Part 3 of this book), DAX is a collection of functions and operators that you can combine to create formulas and expressions for performing advanced data analysis and reporting. It allows users to manipulate and analyze data within Power BI, enhancing reporting capabilities.

Microsoft Power Platform applications allow access to almost every enterprise relational database player in the market; not just Power BI, but also SQL Server, Oracle, DB2, MySQL, PostgreSQL, Sybase, and Amazon Redshift, to name a few.

Power Apps

The cornerstone of Power Platform is the Power Apps tool, which enables users to build custom apps without requiring traditional coding. By using the Power Apps tool, you can speed up the delivery of customized applications at a fraction of the cost of traditional development because you don't need a team of data and design experts. I talk about Power Pages, the external portal functionality that you can derive from Power Apps, in the section "Power Pages," later in this chapter, and in Part 2 of this book.

You can use Microsoft Power Apps to design forms in applications such as SharePoint (Microsoft's online collaboration and content management platform) and Microsoft Dynamics 365 (the suite of enterprise resource planning [ERP] and customer relationship management [CRM] applications). It's also a standalone application in the Microsoft Power Platform, where you can design forms to capture and manage data, automate processes, or integrate with other systems. Power Apps allows you to rapidly develop custom applications or, to a lesser degree, forms in other Microsoft enterprise applications, without requiring much code.

You can use Power Apps to create two types of apps:

- **Canvas apps:** Allow a designer to start with a blank canvas and craft a highly graphical interface by bringing icons, blocks, text fields, labels, and interactive components onto the screen. You drag and drop the components onto the page to create the layout that you want for the application, whether you make it highly sophisticated or as simple as the Compensation Calculator shown in Figure 1-2.

FIGURE 1-2: The Compensation Calculator application created with the canvas app.

When creating a canvas app, you need to be familiar with how to configure the parameters for the elements so that they display the output you want. The calculator illustrated in Figure 1-2 has three data entry elements: OTE (meaning *on-target earnings*), Incentive %, and Multiple (by how many times what the rep brings in exceeds the rep's salary; for example, 10 times their base salary). Based on the data entered, the three calculated fields or columns (Salary, Sales Target, Sales Target Goal) display the output for a sales rep's compensation.

» **Model-driven apps:** Uses a structured data source, such as Dataverse, exclusively to create a form-based experience. You can't change the design unless you use custom controls or JavaScript (which requires some coding experience). That rigidity allows for a consistent look and feel, but also better data quality because the inflexible data structure allows for any user — whether end-user, power user, or developer — to manipulate the data in complex scenarios. If you want to streamline data for business operations, enhance productivity, and focus on data quality by using a consistent form and view-based approach, use model-driven applications.

The form in Figure 1-3 is a highly structured form requiring a user to enter very specific data for an Active Job Opportunity. You could format some fields as drop-downs and allow others to require text entry. The very nature of the data entry allows for repeatable analysis if someone wants to conduct data analysis later on by using a data analytics application, such as Power BI. Figure 1-4 displays a synopsis of two form inputs, presenting the view of the model-driven application for the two job opportunities that a user entered in the system (which an HR professional might do).

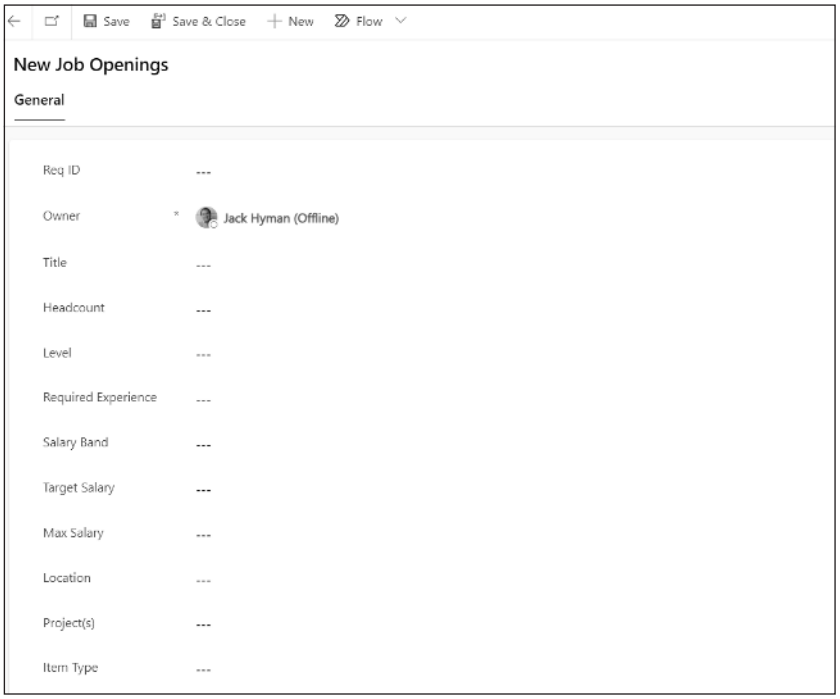

FIGURE 1-3: A model-driven app form created by using Power Apps.

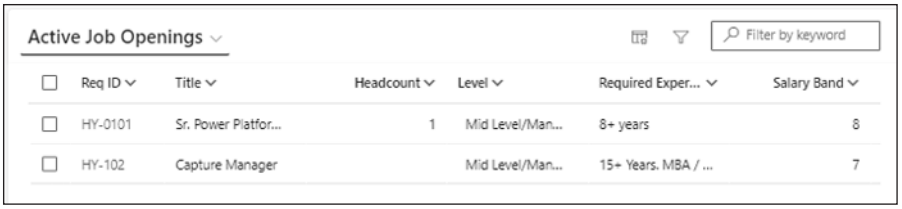

FIGURE 1-4: The view created by a model-driven app in Power Apps.

Dataverse

Microsoft has built all the Power Platform applications around one or more data platforms, with Microsoft's Dataverse being the most commonly used. Dataverse, formerly known as Common Data Services, combines the best of many database platforms that you're probably familiar with, such as Microsoft Access, SQL Server, and even non-relational databases such as NoSQL.

Microsoft Dataverse isn't a relational database; rather, it's a cloud-based storage platform that allows users to store their data and digital assets from two major Microsoft enterprise platforms: Microsoft Power Platform and Dynamics 365 (which I talk about in Chapter 3). Dataverse provides a unified and scalable service-and-app platform where users can securely store and manage their data across business applications. You can create a variety of data solutions, such as tables, views, and form types, without being an expert database administrator or having the infrastructure setup know-how. Microsoft provides Power Apps users with a handful of pre-built example Dataverse tables, such as the Account Table shown in Figure 1-5.

FIGURE 1-5: An example of a Dataverse table.

TIP

The knowledge that you need to set up a Microsoft Access database is about the same as what you need to work with Dataverse, except that Dataverse requires far more structure.

TECHNICAL STUFF

Dataverse is a conglomerate of Microsoft's best database technologies. It stores data types in

- **Azure SQL Server:** A cloud database service for storing and managing relational data in tables
- **Azure Storage:** A cloud service for storing different types of data, such as files, *blobs* (Binary Large Objects), and tables
- **Cosmos DB:** A non-relational (NoSQL) database that scales globally and provides fast access to data
- **Azure Data Lake:** A cloud storage service designed for storing large amounts of structured and unstructured data for analysis
- **Cognitive Search:** An AI-powered service that processes and analyzes large datasets (including those from Azure Data Lake and Dataverse) by using machine learning models to improve data indexing, retrieval, and insights

Power BI

If you're working with data-driven processes, you need to understand and accurately visualize data, as well as transform and model that data so that the results are clear and accurate. *Visualizing* data means showing it in charts or graphs, while *modeling* data means organizing it into tables and successfully connecting those tables together, if appropriate, by establishing relationships.

Power BI provides the comprehensive analytics toolset that you need to create reports and dashboards that can help you uncover insights. But, before creating those dazzling visualizations, you can model and transform data by importing it; cleaning and reshaping it in Power Query (which you can read about in Chapter 9); creating relationships between tables; using DAX for calculations; and applying transformations, such as aggregations and pivots, to prepare it for analysis and visualization.

What makes Power BI much different than Excel is the ability to integrate with other Power Platform components, of course, as well as with more than 100 external data sources that aren't Microsoft branded. Power BI can help your organization foster a data-centric culture. You can get a zest of what Power BI is in Part 3 of this book — but to get a full crash course, check out *Microsoft Power BI For Dummies*, by yours truly (Wiley).

You can use Power BI to analyze and visualize data, whether you collect it in Power Apps or provide it from other data sources. A single user can evaluate insights of

one dataset, derived from an online data source or a simple CSV or XLS file, by using Power BI Desktop. That same user can also access the report produced and its accompanying data when published and shared across an organization through Power BI Services. If you want to surface the data outside of Power BI, you have plenty of options, as well. The most common involves using Embedded Power BI reporting in applications such as Power Apps, or using the Azure-based Power BI embedded data service.

With one or more of the Power BI offerings, users can quickly transform data from simple or complex data models into interactive reports, dashboards, and key performance indicators. Power BI has several options to choose from:

- **Power BI Desktop:** A free desktop application that allows you to connect to a data source, either on a user's desktop or online, to perform data transformations, define the underlying data model, and design reports that visualize insights. End-users can create and analyze most reports independently, enabling them to share and collaborate with others efficiently.
- **Power BI Services:** The online SaaS (Software as a Service) offering of Power BI includes various licensing options, such as Professional and Premium, with the primary differences being the number of dataset refreshes allowed per hour and the storage capacity available per user. You can use Power BI Services to share, collaborate, and distribute reports and dashboards across an organization in a cloud environment.
- **Power BI Report Server:** Unlike Power BI Online, Power BI Report Server is an on-premises solution that allows you to store, manage, and distribute Power BI reports, paginated reports, mobile reports, and traditional SQL Server Reporting Services (SSRS) reports. Report Server also provides a secure environment for hosting reports locally, without relying on the cloud, while offering the flexibility to transition to the cloud with Power BI Service if and when you want to.

Power Automate

Power Automate allows you to create autonomous workflows between apps, data, and third-party services in Power Platform, which can help you streamline business processes and reduce manual tasks. You can use Power Automate for simple notifications and data collection tasks, or to address complex business processes involving multiple steps and conditions. And you don't need a computer science degree to do any of this — Power Automate facilitates the entire process. You simply need to understand the logic behind a business process and data target that flows through the system.

You might also hear Power Automate referred to as a *robotics process automation platform.* That's accurate, as well. With Power Automate, you can create automated workflows between applications and services, in support of data synchronization, file synchronization, and alert notifications; and you don't even have to do any coding. The primary function of Power Automate is to reduce coding for repetitive tasks and processes so that individuals and organizations can more efficiently automate routine operations across applications, inside and outside of the Microsoft ecosystem. Flows come in many varieties, as noted in Table 1-2.

TABLE 1-2 **Flow Types Available in Power Automate**

Flow Name	What It Does
Cloud Flows	Creates a triggered event, such as the arrival of an e-mail from a specific person or the mention of a person in a social media posting.
Instant Flows	Initiate a flow based on a user's interaction with a button. You can automate this type of flow for repetitive tasks that a user performs from a desktop or mobile device.
Scheduled Flows	For activities based on a schedule, such as a daily data upload. For example, you might use this flow to upload a file to SharePoint from a defined location.

You may also hear about two other types of flows in Power Automate: Desktop Flows and Business Process Flows. Desktop Flows focus on automating tasks that occur on the web or directly on a desktop, responding almost instantly to user inputs. Business Process Flows, on the other hand, are structured around predefined steps that guide users through a series of tasks to achieve specific outcomes, ensuring consistency and adherence to processes.

For more about Power Automate, check out Part 4 of this book.

Power Pages

Microsoft Power Pages is a low-code (see the section "Low-code platforms," earlier in this chapter), secure, Software as a Service (SaaS) platform designed for creating, hosting, and managing external-facing business websites. Formerly known as Power Apps Portal, Power Pages provides tools that simplify the web development process so that you don't need to know web-based programming languages to design, configure, and deploy websites that are compatible with various web browsers and devices.

Power Pages includes customizable templates, a design studio for visual editing, and an *integrated learning hub* (a how-to set of educational tools so that you can build that killer website), which together support the efficient creation of websites tailored to specific business needs. You can also integrate other Microsoft Power Platform assets into a Power Pages website, using your existing security credentials from Microsoft Entra (formerly Azure Active Directory).

Power Pages allows for the construction of websites that

- Use shared business data stored in Microsoft Dataverse (see the section "Dataverse," earlier in this chapter, for a discussion of this storage platform)
- Develop built-in apps through Power Apps (see the section "Power Apps," earlier in this chapter)
- Create workflows in Power Automate (flip back to the preceding section)
- Integrate intelligent virtual agents, which Microsoft refers to as *Copilots*
- Draft reports by using Power BI (check out the section "Power BI," earlier in this chapter).

The integration of Power Pages with Microsoft Dataverse facilitates a unified development process and ensures data consistency across various applications and services. You can build dynamic, data-driven websites that can support various external business functions through credential-based access.