**Slide 1: Power BI: Fundamentals, Optimization, and Comparison**

**Key Insights for New Users and Beyond**

**Slide 2: What is Power BI?**

* **Interactive Data Visualization Software**
  + A **Microsoft-powered business intelligence (BI) tool** designed to transform various data sources into static and interactive data visualizations.
  + Helps turn raw data into **interactive dashboards and actionable insights**.
* **Core Capabilities**
  + Allows users to **connect to hundreds of sources** (e.g., Excel, SQL databases, CSVs, JSON, webpages, PDFs).
  + Enables **cleaning and shaping data** with minimal coding.
  + Facilitates **creating custom calculations** and visualizing data through charts, graphs, and reports.
  + Supports **efficient data exploration** through hierarchies and drill-down paths.
  + **Directly impacts performance**, leading to faster query execution when designed well.
  + **Promotes accurate, consistent, and dependable reports**.
  + Ensures **easier future maintainability** by creating reusable components and standardizing data.
* **Widespread Adoption**
  + Used in industries for data storytelling, decision-making, and analytics.
  + Integrates well with tools like Excel, databases, and cloud services.
  + **Over 3,000 companies use Microsoft Power BI** for business intelligence.

**Slide 3: Power BI Key Components**

* **Power BI Desktop**
  + **Free Windows desktop application** for authoring and designing reports.
  + Includes **Power Query** (for data getting, cleaning, transformation) and **Power Pivot** (for data modelling) tools.
* **Power BI Service**
  + A **cloud-based, Software as a Service (SaaS) online platform**.
  + Used for publishing reports, collaboration, and sharing insights.
* **Power BI Mobile Apps**
  + Available for Android, iOS, and Windows devices, enabling access to reports and dashboards on the go.
* **Power Query**
  + Facilitates ETL (Extract, Transform, Load) processes.
  + Provides a **Graphical User Interface (GUI)** for common data preparation without coding.
  + Uses the M formula language for advanced operations.
* **DAX (Data Analysis Expressions)**
  + The **formula language for creating custom calculations**.
  + Enables creating new information from existing data in your model.
* **Power BI Gateway**
  + Used to **sync external data** and enable automated refreshes, especially for on-premises data sources.
* **Power BI Premium**
  + A **capacity-based offering** providing greater scale, performance, and advanced features (e.g., Dataflows, Paginated Reports, AI) without requiring individual user licenses for recipients.
* **Power BI Dataflow**
  + A **cloud-based Power Query implementation** for data transformations to create a common Power BI Semantic Model.
* **Power BI Semantic Model** (formerly Dataset)
  + A collection of data for use in Power BI reports, which can be connected to or imported.

**Slide 4: Data Modeling Fundamentals in Power BI**

* **Significance of Data Modeling**
  + **Process of defining data structure, properties, and relationships** within a data model.
  + A **logical representation** of how data is structured and related within the tool.
  + Has a **major impact on report performance**.
* **Star Schema: A Best Practice**
  + A **well-established data modeling method** commonly used in relational data warehouses.
  + Requires categorizing tables as either **dimensions** or **facts**.
  + **Considered a best practice for Power BI** due to enhanced performance, flexibility, easy maintenance, and improved usability.
  + **DAX is optimized for star schema**; it generally outperforms single flat tables in performance, RAM, and storage efficiency.
  + A **common mistake** for new users is not learning or using a star schema.
* **Fact Tables**
  + Store **quantitative, numerical data that can be aggregated** and analyzed.
  + Typically contain information about events or transactions (e.g., quantity sold, revenue).
  + **Can hold a very large number of rows** and continue to grow over time.
* **Dimension Tables**
  + Provide **context and additional details** to the numerical data in fact tables (e.g., product name, category).
  + Generally contain a relatively **small number of rows**.
  + **Linked to fact tables through relationships**.
* **Primary and Alternative (Foreign) Keys**
  + **Primary Key (Surrogate Key)**: A column or group of columns that **uniquely identifies a row** in a table.
  + **Alternative Key (Foreign Key)**: A column in one table whose values correspond to the primary key in another table, forming the basis of a relationship.

**Slide 5: Data Modeling Fundamentals (Cont.)**

* **Cardinality**
  + Refers to the number of unique values in one table related to another.
  + **One-to-One (1:1)**: One record in the first table relates to one in the second. **Rare** in Power BI data modeling.
  + **One-to-Many (1:N)**: One record in the first table relates to many in the second. This is the **most common type** in Power BI, linking a dimension table (one side) to a fact table (many side).
  + **Many-to-Many (N:N)**: Many records in the first table relate to many in the second. Not directly supported and infrequently used.
* **Cross-Filter Direction**
  + Determines the direction(s) filters propagate between tables.
  + **Single**: Filters propagate in **one direction only**. Generally preferred.
  + **Bi-directional**: Filters propagate in **both directions**. Less common, can negatively impact performance, and create ambiguous filter paths.
  + 1:1 relationships can *only* be bi-directional, while 1:Many and Many:Many can be single or bi-directional.
  + **Avoid bi-directional relationships against high-cardinality columns or excessive use**.
* **Active and Inactive Relationships**
  + By default, relationships are **active**.
  + **Only one active relationship can exist between two tables at a time** to avoid conflicting or incorrect results.
  + To use **inactive relationships**, you can define a DAX expression with the USERELATIONSHIP function or duplicate the dimension table.
    - Duplicating tables increases model size but allows simultaneous filtering.

**Slide 6: DAX: Data Analysis Expressions**

* **What is DAX?**
  + A **collection of functions, operators, and constants** used in formulas to calculate and return one or more values.
  + Helps create new information from data already in your model.
* **Importance**
  + Enables complex calculations like growth percentage or year-over-year growth.
  + Essential for getting the most out of your data and solving real business problems.
* **Key Concepts**
  + **Syntax**: How the formula is written, including measure names, operators (e.g., =), DAX functions (e.g., SUM, CALCULATE), parentheses for arguments, and references to tables and columns (e.g., Sales[SalesAmount]).
  + **Functions**: Predefined formulas that perform calculations.
    - Always **reference a complete column or a table**, not individual cells like Excel's VLOOKUP.
    - Can include categories like Date and Time, Time Intelligence, Logical, Statistical, and Text.
    - DAX formulas can contain up to **64 nested functions**.
  + **Context**: One of the most important DAX concepts.
    - **Row Context**: Refers to the **current row** when a formula applies filters to identify a single row.
    - **Filter Context**: One or more filters applied in a calculation that determines a result or value. It **applies in addition to row context**.
    - Can be set by adding fields to a visualization or through DAX functions like ALL, RELATED, FILTER, or CALCULATE. The CALCULATE function is particularly powerful for modifying filter context.
* **Measures vs. Calculated Columns**
  + **Always use explicit measures for values**; do not use implicit aggregates.
  + **Avoid creating calculated columns if they are not necessary**. If needed, add them in Power Query (which is generally better for performance).

**Slide 7: Performance Optimization & Common Mistakes**

* **Prioritize Optimization**
  + It's often recommended to **worry about performance optimization only when you have a working solution that needs it**.
* **Data Preparation (ETL: Extract, Transform, Load)**
  + **Do your transformations as far upstream as possible** (e.g., SQL query, then Power Query, DAX as a last resort). This is crucial for performance.
  + **Combine/append different fact tables into one main fact table**.
  + **Filter out/remove any columns that won't be used** in Power Query to reduce file size. Using "Choose Columns" is often better than "Delete Columns" for flexibility.
  + **Always aggregate when possible** to reduce data size. For monthly data, use a monthly data table and aggregated data instead of a daily date table.
  + **Reduce precision** (number of decimals) and **convert numbers to integers**.
  + **Push calculations down the BI stack**: Perform them in Power Query or the source database before DAX.
* **Data Model Design**
  + **Implement a clean star schema**. This is a must for optimal performance.
  + **Build a semantic model for data, not just for a specific report**. This is a more advanced concept and may not be ideal for beginners' initial learning.
  + **Use a dedicated date table**, including business days, months, weeks, first/last day of month, etc..
  + **Deactivate/remove Power BI's auto date-time feature**.
  + **Hide foreign keys (IDs) from report view** so fact tables only show explicit measures.
  + **Think about reusability**. Moving transformations into dataflows can mean changes only need to be done once across multiple reports.
  + **Avoid complex DAX formulas** if simpler alternatives are available.
  + **Avoid excessive bi-directional or many-to-many relationships**. Many-to-many relationships should ideally be single direction.

**Slide 8: Performance Optimization & Common Mistakes (Cont.)**

* **Connection Type**
  + **Prioritize Import mode** over Live Connection, Composite, and DirectQuery for better performance.
  + **Avoid DirectQuery (direct connection)** unless dealing with huge datasets that require constant real-time refreshes. DirectQuery can be **4-5 times slower** as each visual interaction triggers a query back to the source.
* **Report & Visual Design**
  + **Reduce the number of visuals per page**.
  + Implement an **efficient report layout pattern** (e.g., Overview > Details > More details).
  + **Scrolling down on reports indicates too much information is crammed onto one page**; consider using buttons to switch views instead.
  + **Aim for an update time of less than 2 seconds** when filtering; slower response times deter users.
  + **Apps are the best way to manage reports and dashboards** with access control.
  + **Presentation is more important than functionality** for gaining recognition.
  + **Simple is better than powerful and complicated**.
  + **Turn off "detect relationships on load"** in Power BI options and save as a template.
  + **Export and save themes** after finishing a report to reuse in new templates.
  + Consider using tools like "Measure Killer" for cleanup.

**Slide 9: Data Visualization Best Practices**

* **Embrace Clarity**
  + Avoid unnecessary formatting, clutter, or elements that distract from the data.
  + **Label data points and use clear titles on charts and graphs**.
  + Avoid drop shadows and 3D formatting, as they add no value to the visual.
* **Be Consistent**
  + **Keep design elements uniform** (e.g., color, line weight) across your visuals.
  + Ensure the **same name and color represent a particular data point** across all related charts and dashboards.
* **Support Readability**
  + Use **light and dark values to create contrast** instead of relying solely on different colors, to assist users with low vision or color blindness.
  + Use **easy-to-read font types, sizes, and text orientation**. Stick to **sans-serif fonts** (e.g., Arial, Calibri).
  + Limit the number of fonts and font sizes to three per report.
  + Choose **shape fills and backgrounds that support, not obscure, numbers and text**; avoid distracting patterns.
* **Achieve Balance**
  + Leave appropriate "white space" between chart elements.
  + Show the right number of data relationships per chart, generally **capping at six per chart or graph**. If there are too many, break them into multiple charts.
  + Avoid crowding visuals.
* **Get Input**
  + Seek feedback on your visualizations from colleagues or your audience to ensure your message is clear and understood.
* **Pursue Truth**
  + **Choose the chart type that best represents the data** (e.g., pie charts for proportion comparison, line charts or combination charts for showing trends).
  + **Start from zero on the Y-axis for graphs** to avoid misinterpretation, unless there is a clear and justified reason not to.

**Slide 10: Power BI Security**

* **Foundation**
  + Built on **Azure**, Microsoft's cloud computing platform.
  + Utilizes **Microsoft's strong security stack**, earning high security classifications.
  + Trusted by national security agencies, financial institutions, and healthcare providers for sensitive information.
* **Authentication**
  + User authentication follows **Azure Active Directory's (Azure AD) auth code grant flow**.
  + All communication between the browser and Power BI service uses **HTTPS and TLS 1.2 or above**.
  + Azure Traffic Manager routes users to the nearest datacenter for optimal performance.
  + Azure AD security tokens are used for authentication and session maintenance.
* **Data Residency**
  + Customer data is stored in an Azure geography assigned during tenant setup, known as the "home geo".
  + **Multi-geo deployments** allow data to be stored in different Azure geographies to comply with local regulations, though some metadata may remain in the home geo.
* **Data Handling**
  + **Data at rest is encrypted by default** using Microsoft-managed keys (Azure SQL's Transparent Data Encryption, Azure Storage Encryption).
  + Organizations with Power BI Premium can utilize **"bring your own key" (BYOK)** to encrypt imported data with their own keys.
  + Actively processed data in memory **is not encrypted**.
  + Data in transit is **encrypted using HTTPS**.
* **Authentication to Data Sources**
  + For **imported data**, Power BI uses the user's credentials established during connection; viewing reports **does not access the underlying source** once imported.
  + For **DirectQuery**, credentials (pre-configured or current user's via single sign-on) are used to connect to the source **each time data is viewed**. This allows for **Row Level Security (RLS) and Object-Level Security (OLS)** implemented at the data source.
    - **RLS** restricts data access at the row level based on user roles and filters.
    - **OLS** secures sensitive tables or columns by obscuring their names and metadata from unauthorized users.
* **Data Loss Prevention (DLP)**
  + Deep integration with **Microsoft Information Protection (MIP) sensitivity labels** allows classifying and labeling sensitive data.
  + Governance policies can be enforced even when content is exported (e.g., Excel, PowerPoint, PDF, .pbix files).
  + **Microsoft Defender for Cloud Apps** monitors and controls risky Power BI sessions, blocks sensitive data downloads, and detects suspicious user activity.

**Slide 11: Power BI vs. Tableau Comparison: Key Differences**

* **Ease of Use**
  + **Power BI**: User-friendly interface with easy integration into the Microsoft ecosystem. Considered **easier for new users** who aren't data analysts.
  + **Tableau**: Intuitive drag-and-drop interface but a **steeper learning curve for advanced features**. Popular with seasoned data analysts.
* **Data Integration**
  + **Power BI**: Excellent integration with **Microsoft products** (Excel, SharePoint, SQL Server, Azure, Dynamics 365) and a variety of other data sources.
  + **Tableau**: Connects to a **wider range of data sources**, including cloud-based databases, web services, and third-party applications.
* **Performance**
  + **Power BI**: Good performance, optimized for Microsoft Azure cloud. **Faster with limited data volume**, but can slow down when handling bulk data.
  + **Tableau**: Strong performance with **large datasets**, though it can be resource-intensive. Provides extensive features for visualizing data and does not limit data points or enforce row/size limitations.
* **Cost**
  + **Power BI**: More **affordable pricing**, including a free version and low-cost Pro licenses (under $10/month per user for Pro, $4,995/month for Premium).
  + **Tableau**: Higher cost (Pro version over $35/month per user, Creator $70/month, Viewer $12/month).

**Slide 12: Power BI vs. Tableau Comparison (Cont.)**

* **Visualization Capabilities**
  + **Power BI**: Robust visualization capabilities, offering a wide variety of charts, maps, and KPIs. Slightly less flexible than Tableau for extensive customization.
  + **Tableau**: **Extensive and highly customizable visualization options**, with a wide range of chart types and interactive dashboards.
* **Collaboration**
  + **Power BI**: Good collaboration features, especially with integration into Microsoft Teams. Offers co-authoring and commenting built-in.
  + **Tableau**: Good collaboration features with Tableau Server and Tableau Online, but may require third-party tools for some functionalities.
* **Artificial Intelligence (AI) & Programming Support**
  + **Power BI**: Built-in AI capabilities like **Q&A, AI visuals, and integration with Azure ML**. Some advanced features like AutoML and Cognitive Services are only in Premium workspaces. Supports DAX and M language; can connect to R (mostly for enterprise users).
  + **Tableau**: Advanced AI features with Tableau AI and integration with R and Python. Integrates better with R; SDK can be used with C, C++, Java, and Python. Does not require coding for basic visualizations.
* **Community & Support**
  + **Power BI**: Strong community support with robust Microsoft resources. Relatively younger in the market, so a smaller community than Tableau.
  + **Tableau**: **Large, active user community** (over 160,000 active users) with extensive online resources and professional support.
* **Decision Factors**
  + The choice ultimately depends on the **specific needs of the user and organization**, including existing Microsoft ecosystem integration, data complexity, and customization requirements.
  + Learning both tools can be beneficial to expand skillset and increase opportunities.