RPM, AL CODE, & POWER BI:
A SYNERGISTIC APPROACH

MAXIMIZING PROFITABILITY & ENHANCING CUSTOMER EXPERIENCE FOR EQUIPMENT DEALERSHIPS.

In modern resource management and data analysis, the convergence of Suite Engine's RPM, AL code, and Power BI presents a potent synergy that unlocks comprehensive insights and optimization opportunities. This document introduces the interconnected world of RPM, AL code, and Power BI, highlighting their collaborative role in resource utilization analysis.

RPM

At the heart of efficient resource management lies RPM, a dynamic framework that orchestrates and streamlines rental operations. RPM empowers organizations to handle complex workflows, monitor asset availability, manage contracts, and ensure smooth customer interactions. This section shows how RPM's structured data forms the foundation for analytical exploration and how AL code bridges the gap between this data and Power BI.

AL CODE: BRIDGING DATA TO INSIGHTS

AL code emerges as a critical bridge between RPM's raw data and the actionable insights sought through Power BI. AL coding allows the integration of Business Central data into Power BI, enabling advanced data analysis. Through Tables, Code Units, API Pages, and API Queries, AL code structures and aggregates data, making it accessible via a dedicated Dataflow within a Power BI Service Workspace. This integration enables seamless data transformation, modeling, and eventual visualization in Power BI Desktop.

POWER BI: ILLUMINATING INSIGHTS

Power BI steps into the spotlight as the visualization powerhouse that transforms aggregated data into meaningful insights. To utilize many of Power BI's features, this document operates on the assumption that a Power BI Pro license is available for both development and end-user consumption. This assumption sets the stage for robust collaboration, data exploration, and decision-making across the organization. By aligning the worlds of RPM, AL code, and Power BI, organizations can extract valuable insights from their rental operations, make informed decisions, and drive operational excellence.

UNIT TIME UTILIZATION IN RPM

In RPM, Unit Time Utilization is a crucial metric for assessing resource efficiency within a specific time frame. Unit Time Utilization is computed as Days On-Rent divided by Days Available in a given period. Days On Rent for a unit on a contract line is Stop Date minus Start Date. The Stop Date is the Actual Stop Date, or if absent, the Estimated Stop Date.

In RPM, contract lines exist as RPM Line or RPM Line Archive entries. Data is chosen based on conditions. If a unit is in RPM Line, its data comes from RPM Line entries. If in both the RPM Line and the RPM Archive Line, the RPM Line data is used. If in RPM Archive Line only, the latest RPM Line Archive version is employed.

Fundamentally, Unit Time Utilization in RPM involves adhering to conditions, specific data sourcing, transformation, modeling, and intricate data analysis. This orchestrated process yields an accurate gauge of resource deployment efficiency.

DURATION

Unit Time Utilization is calculated by dividing Days On-Rent by Days Available within a specific time frame. Data retrieved from contract lines using AL code facilitates the computation of Days On-Rent. This can be categorized as “duration,” indicating the period between a unit’s Start Date and Stop Date on a contract line.

THE ROLE OF AVAILABILITY & LOCATION IN RPM

Availability and Location are pivotal elements within the RPM system, integral for efficient resource management and precise Unit Time Utilization calculations.

AVAILIBILITY

This signifies a unit’s contract readiness. Activation, triggered upon adding a unit to Fixed Assets, marks its eligibility for contract placement. Conversely, Deactivation denotes unavailability for contracts. The computation of Days Available involves assessing the feasible rental time frame and aggregating available days within the specified period.

LOCATION

Designates the contract-placement site or “responsibility center.” Accurate tracking ensures precise Time Utilization calculations for each unit. This tracking also facilitates evaluating the operational efficiency of responsibility centers based on Unit Time Utilization, thereby enhancing management and decision-making processes.

PROCESSING RPM UNIT MOVEMENT LEDGER ENTRIES FOR UNIT TIME UTILIZATION ANALYSIS

RPM Unit Movement Ledger Entries permit multiple entries on any given day. In the context of Unit Time Utilization for Power BI, our focus is on location, indicating availability for contract usage. This involves identifying a unit's location on a specific day for reporting purposes. To achieve this, the AL code examines whether an Entry Type equals Activation or Deactivation, which signifies a unit's availability's initial and terminal points.

Furthermore, when multiple entries exist on a particular date, the AL code will prioritize an entry with an associated contract (Source Document No.) if available. If no such contract-related entry is present, the AL code will choose the Entry with the last Entry No., for the unit on that date.

The ultimate goal is to have the data primed for modeling and further manipulation using Power BI's DAX in preparation for generating visualizations that enhance reporting.

ETL IN THE CONTEXT OF POWER BI

ETL stands for **Extract, Transform, and Load**. It's a process used in data management and analytics to prepare and integrate data from various sources for analysis, reporting, and visualization. ETL plays a vital role in transforming raw data into a structured, clean, and meaningful format that can be used for business intelligence and decision-making. It is important to note that ETL can be split across Power BI Service and Power BI Desktop.

1. EXTRACT

Data is initially extracted from diverse sources, such as databases, spreadsheets, APIs, and web services. Power BI provides a hassle-free connection to various data sources, allowing you to pull relevant data into your workspace.

2. TRANSFORM

Extracted data undergoes essential steps to make it analysis-ready:

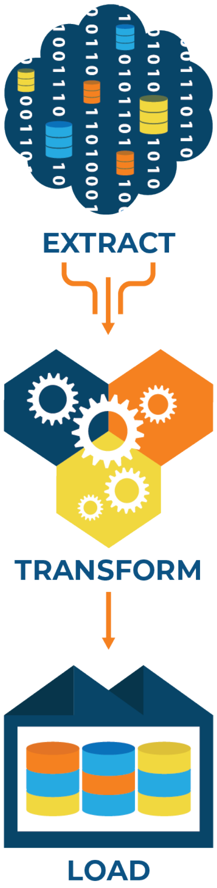
- **GROUPING:** Using Power Query's grouping and aggregation functions, data is manipulated to allow the AL code logic to be appropriately filled across the dataset.
- **JOINING & MERGING:** Tables from various sources are combined, forming a unified dataset for comprehensive insights.
- **DATA CLEANING:** Improving data quality involves eliminating duplicates, filling missing values, and correcting errors. Importantly, errors caught during the AL code or data cleaning process can provide valuable insights for process improvement, preventing similar issues in the future.

  Organizations can refine their data collection, Entry, and validation processes by analyzing the nature and frequency of these errors, ensuring higher data accuracy over time. This proactive approach improves the quality of current analyses and establishes a continuous feedback loop for data optimization.

- **CONDITIONAL TRANSFORMATIONS:** Data adjusts based on conditions, like filtering rows or applying custom calculations. Utilizing M-Code, you derive meaningful insights from existing columns. Power BI's powerful features in the Transform phase enable visual or M-Code-driven actions, ensuring data refinement for subsequent analysis.

3. LOAD

Transformed data is loaded into Power BI's data model, employing an in-memory columnar database for efficient storage and management. This data model is the foundation for creating reports, dashboards, and visualizations.



DAX IN POWER BI: BUILDING DYNAMIC ANALYSIS & INSIGHTS

DAX, or **Data Analysis Expressions**, is a formula language in Power BI for calculations, custom tables, and measures. It powers dynamic data analysis and insights within Power BI, working alongside the data model, measures, and visualizations. DAX's roles in Power BI:

1. CREATING ON-THE-FLY TABLES

DAX enables the creation of custom tables that act as snapshots for focused calculations and analysis. Think of them as lenses that zoom in on specific aspects of your data, enabling precise exploration.

2. CALCULATED COLUMNS

DAX enables the creation of calculated columns within your data model. Calculated columns are derived from existing columns in your dataset, applying specific calculations or logic. These columns are computed during data import and become part of the data model. Calculated columns are suitable for scenarios where you must extend the dataset with additional calculated values that remain constant throughout the analysis.

3. MEASURES

Measures are at DAX's core, dynamically calculating aggregated data. Unlike calculated columns, measures adjust with user interactions in visualizations. They offer calculations like sums, averages, percentages, ratios, and complex functions. Measures drive real-time insights and Key Performance Indicators (KPIs).

DAX LINKS THE DATA MODEL, MEASURES, & VISUALIZATIONS

DATA MODEL

DAX uses data model relationships for table calculations and aggregations. This model enables intricate calculations with multiple tables and dimensions, enriching analysis depth.

MEASURES

DAX expressions, written for “measures,” are dynamically calculated based on user interactions. Measures pull data, apply calculations, and they provide context-aware results for visualizations. They can be displayed in tables, charts, and more.

VISUALIZATIONS

DAX measures seamlessly integrate into visualizations, delivering dynamic insights in charts, graphs, and tables. Measures adapt to user interactions, responding to filter, slice, and drill-down changes

DAX IN SUMMARY

DAX is a potent formula language within Power BI, enabling users to craft custom tables, calculated columns, and measures. It collaborates with the data model and visualizations, equipping users to conduct advanced calculations, steer data-driven decisions, and present dynamic insights to stakeholders.

MAXIMIZING INSIGHTS THROUGH STRATEGIC DATA ELEMENTS & PRECISE MEASURES

The architecture of the _rpmFT dataset and its corresponding data model revolves around the pivotal concept of Unit Time Utilization. However, the potential for refined insights is broader than this foundation alone. An opportunity exists to enrich Unit Time Utilization by harnessing additional data elements.

For instance, insights-rich visuals are created by blending meticulously crafted calculated columns, well-defined measures, and skillful page and visual filters. These include but are not limited to branch time utilization YTD, time utilization by month current year, time utilization by month prior year, total company utilization YTD, CM forecast, and utilization and forecast by day.

Refined visualizations offer a deeper dive into the dataset, unraveling nuanced patterns and trends. They transform data into actionable insights, empowering users to glean unparalleled value from the dataset.indicating the period between a unit’s Start Date and Stop Date on a contract line. And, of course, the possibilities continue beyond Time Utilization alone. The dataset is a treasure trove of potential insights.

Talk to your Account Executive to discuss your next step to tapping into the data realms residing within RPM.

GET YOUR PERSONALIZED DEMO