CANalitica

Project Management Business Intelligence (PMBI)

User Guide V1.0

August 2026

CANalitica Professional Services Inc.

CANalitica

Acronyms and Terminology

Term / Acronym	Definition
AC	Actual Cost. The total costs incurred for work performed on a project up to a given point in time. Also referred to as Actual Expense in PMBI.
BAR	Budget Allocation Ratio. A metric indicating the proportion of the total authorized budget that has been assigned to tasks by way of Planned Value. A value of 1.00 means the full budget is allocated.
Baseline	The original approved plan for a project, including scope, schedule, and budget, against which performance is measured.
Commit / Committing	A reserved term in PMBI indicating that data will be permanently written to the application database. This applies to both user configuration changes and project data snapshots.
CPI	Cost Performance Index. A measure of cost efficiency calculated as Earned Value divided by Actual Cost. A value below 1.00 indicates a cost overrun.
EAC	Estimate at Completion. The current forecast of the total cost to complete a task or project, based on actual performance to date.
EV	Earned Value. The budgeted value of work actually completed at a point in time, expressed in dollars.
EVM	Earned Value Management. A project controls methodology that integrates scope, schedule, and cost to objectively measure project performance and progress.
KPI	Key Performance Indicator.
LOE	Level of Effort. An earning technique used for support activities where value is tied to the passage of time rather than the production of a discrete deliverable.
Mapping	The process of linking columns in a user's source data file to the corresponding fields in the PMBI data model.
PCO/CO	Project Change Order or just Change Order. A formal document used in project controls to record a proposed or approved change to project scope, budget, or schedule.
PMBI	Project Management Business Intelligence. The name of the CANalitica application described in this guide.
PMO	Project Management Office. An organizational unit responsible for standardizing project management practices and overseeing a portfolio of projects.
PV	Planned Value. The budgeted value of work scheduled to be completed in a given point in time, according to the approved project plan.

Re-baseline	A formal revision to the project's approved performance measurement baseline, typically in response to significant scope, schedule, or budget changes.
Reporting Period	A defined time interval, typically a calendar month, for which project performance data is committed and reported in PMBI.
RFI	Request for Information. A formal document used in construction and engineering projects to clarify scope, design, or contract requirements.
Snapshot	A committed record of project performance data for a specific reporting period, written to the PMBI database and used to populate dashboard reporting.
SPI	Schedule Performance Index. A measure of schedule efficiency calculated as Earned Value divided by Planned Value. A value below 1.00 indicates the project is behind schedule.
SV	Schedule Variance. The dollar difference between Earned Value and Planned Value at a point in time. A negative value indicates the project is behind schedule.
Tenant	An organization with its own isolated workspace in PMBI, provisioned upon subscription (select plans only have this option).
VAC	Variance at Completion. The projected budget surplus or shortfall at project completion, expressed as the difference between the current authorized budget and the Estimate at Completion.
VAC Ratio	Variance at Completion expressed as a ratio relative to the current authorized budget, used in PMBI to provide a normalized performance signal on a consistent scale.
WBS	Work Breakdown Structure. A hierarchical decomposition of a project's total scope into progressively smaller, deliverable-oriented components, down to individual work packages.
Work Package	The lowest level of the WBS at which budget is assigned, actuals are recorded, and EVM metrics are calculated.

CANalitica

Table of Contents

Contents

1.0 Introduction

1.1 Purpose

This User Guide provides instructions and reference material for users of **CANalitica PMBI**, a cloud-hosted SaaS application developed by CANalitica Professional Services Inc.

PMBI is designed to help project teams, PMO directors, and executives monitor and communicate project performance through standardized Earned Value Management (EVM) reporting. Built and hosted on Microsoft Azure, the application provides visibility into cost, schedule, and progress metrics across an organization's entire project portfolio without requiring any BI expertise or infrastructure to manage.

This guide covers the following topics:

- **Getting Started** — Subscribing, onboarding your organization, and signing in for the first time.
- **Managing Users** — Adding team members, assigning roles, and controlling access.
- **Uploading & Committing Data** — Preparing your project data, uploading it to the platform, and committing a reporting period "snapshot".
- **Navigating the Dashboards** — Understanding the available report pages, filters, and interactive features.
- **Understanding EVM Metrics** — Key performance indicators, data fields, and how calculations are derived.

Note: images in this document may slightly differ than what is shown in the actual PMBI application.

1.2 Key Benefits

CANalitica PMBI is designed to complement your existing project management processes and tools, not replace them. Your team continues to manage projects day-to-day in the processes and tools they already use. PMBI fits into that workflow by accepting periodic data uploads from your existing sources, turning that data into structured EVM reporting without requiring a separate database to maintain or a new system to learn from scratch.

Key benefits include:

- **Standardized EVM reporting** — Consistent cost and schedule performance metrics across every project in your portfolio, including SPI, CPI, and variance analysis, without custom BI development.
- **WBS hierarchy with drill-down** — Navigate from a portfolio-level view down to individual work packages, giving project managers and executives the right level of detail when they need it.
- **AI-generated health narratives** — After each reporting period, the platform can produce plain-language summaries of project health, so stakeholders can get a clear picture without interpreting raw data themselves *(available on select plans only)*.
- **Fits your existing stack** — Upload data from Excel, CSV, or direct SQL connections depending on your plan. Additional connectors will be added soon.
- **Secure by default** — Sign in with your existing Microsoft account via Microsoft Entra ID. No separate credentials to manage.
- **Cloud-hosted and always available** — Access your dashboards from any device, anywhere, without infrastructure to deploy or maintain.
- **Scales with your portfolio** — Tiered plans mean you start at the right level and grow without switching platforms.
- **Your data, your way** — PMBI does not require you to migrate away from the systems your team already uses. You continue managing your projects in whatever tools and processes work for your organization. You simply provide already existing data periodically for reporting. The source data stays with you. There are no proprietary data formats to adopt, no vendor lock-in, and no dependency on any specific scheduling or project management platform.

1.3 Getting Support

CANalitica is committed to helping you get the most out of PMBI. At this time, support is available through the following channels:

User Guide

This document is the primary reference for all platform features, data preparation guidance, and EVM concepts. Before reaching out, we encourage you to consult the relevant section of this guide, as most questions about platform navigation, data requirements, and reporting interpretation are addressed here.

Email Support

For questions not addressed in this guide, you can reach CANalitica support at pmbisupport@canalitica.ca. We are committed to responding as promptly as possible.

What Support Covers

Email support is intended to assist with matters directly related to the PMBI platform, including:

- Questions about platform features, navigation, and behaviour
- Clarification on data table requirements, column formats, and commit processes
- Troubleshooting unexpected errors or results within the application
- Questions about your subscription, plan limits, or account access
- Providing Outbound IP Addresses
- Guidance on interpreting EVM metrics and report visuals as they appear in PMBI

What Falls Outside of Support Scope

- Preparation, cleaning, and formatting of your source project data (although we can provide some guidance)
- Design or validation of your project's WBS structure
- Configuration of your internal databases, firewalls, or network settings for SQL connectivity
- Project management advice, scheduling guidance, or EVM methodology consulting beyond what is described in this guide
- Issues arising from the source data errors

If your organization requires hands-on assistance with data preparation, WBS design, or EVM implementation, CANalitica Professional Services Inc. offers consulting engagements separately from the platform subscription. Contact us at pmbisupport@canalitica.ca to discuss your needs.

Platform Infrastructure

PMBI is hosted on Microsoft Azure and depends on Azure services for availability, performance, and security. While CANalitica monitors the platform and works to maintain service continuity, infrastructure-level incidents including Azure service disruptions, regional outages, or scheduled maintenance windows are outside of CANalitica's direct control. In the event of a platform outage, CANalitica will communicate status updates as information becomes available.

For real-time Azure service health information, Microsoft publishes current and historical incident reports at https://status.azure.com.

Power BI report rendering is similarly dependent on Microsoft's Power BI service. Degraded report performance or rendering issues that originate from the Power BI service rather than the PMBI platform itself are subject to Microsoft's own service health and support processes.

The AI Project Summary feature depends on third-party AI model providers. Availability of the narrative generation feature is subject to the service health of those providers, and temporary disruptions may result in narrative generation being unavailable or delayed. These events are outside of CANalitica's control. If the AI summary feature is unavailable, all other platform features and report data remain unaffected.

CANalitica is not able to offer compensation, credits, or service level guarantees in connection with third-party infrastructure incidents including AI service providers. Organizations with formal uptime or availability requirements should review the Microsoft Azure and Power BI service level agreements directly, as these govern the underlying infrastructure on which PMBI operates.

2.0 Getting Started

2.1 Subscribing to CANalitica PMBI

CANalitica PMBI is available through the Microsoft Azure Marketplace. To subscribe, visit the Marketplace listing and select the plan that best fits your organization's portfolio size and reporting needs. A free trial is available on select plans.

After selecting your plan and completing the Microsoft checkout process, you will be redirected to the CANalitica PMBI setup page to register your organization (fig 1). It is how your workspace is created and your administrator account is provisioned.

Once submitted, please allow a few minutes for provisioning to take place. Your administrator account will be created for you using the email address associated with your Microsoft purchase which you can use to sign in and add more users.

All subsequent access is through the CANalitica PMBI URL: **https://pmbi.canalitica.ca**.

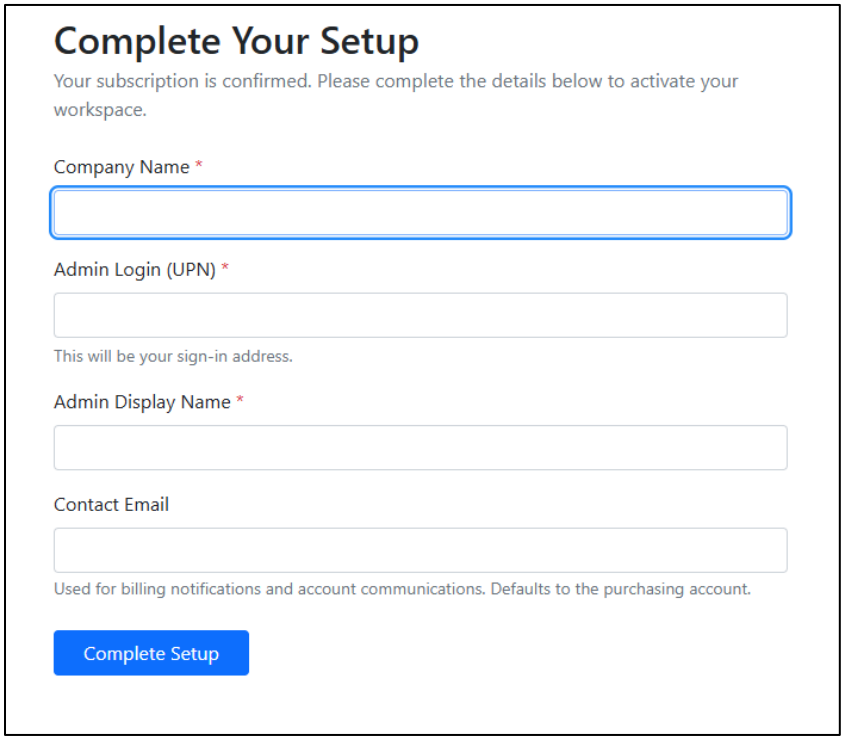


Figure 1: Final step in the subscription process

2.3 Signing In

CANalitica PMBI is accessed by all organizations through a single web address:

https://pmbi.canalitica.ca

To sign in, navigate to the address above. You will be prompted to authenticate using your existing Microsoft account (fig. 2).

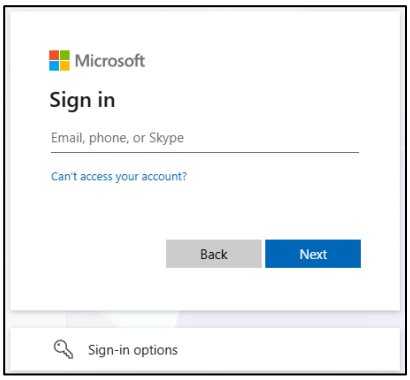


Figure 2: The sign-in prompt

How your workspace is identified:

CANalitica PMBI uses a two-step process to ensure your organization's data is securely isolated from other tenants:

1. Your Microsoft account authenticates your identity via Microsoft Entra ID.
2. The platform then looks up your account in its secure directory to confirm your organization membership, role, and access level. On select plans, your organization is provisioned with a dedicated Power BI dataset, providing an additional layer of data isolation beyond the access controls applied to all workspaces

Your administrator controls who has access by managing users within the workspace settings.

Note: If you receive an access denied message after signing in with a valid Microsoft account (fig. 3), contact your workspace administrator to confirm your account has been added to your organization's PMBI workspace.

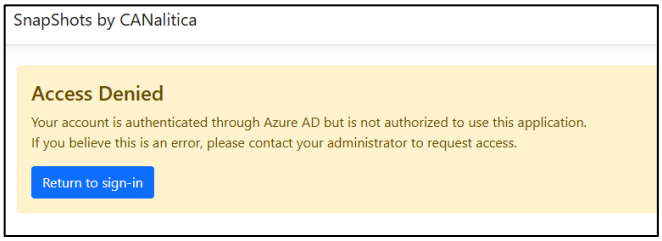


Figure 3: Attempting to log in without an account

3.0 Managing Your Users

3.1 Overview

CANalitica PMBI has two user roles:

Role	Description
Administrator	Full access to the dashboard and reports in addition to user management and data upload functions.
User	Read-only access to dashboards and reports for projects they have been granted access to.

Application management is handled with the Administrator role. Administrators are responsible for adding and managing users and their access, configuring data connections, and creating the period-based reporting for projects. All User management functions are accessed through the **User Admin** page of the application (fig. 4).

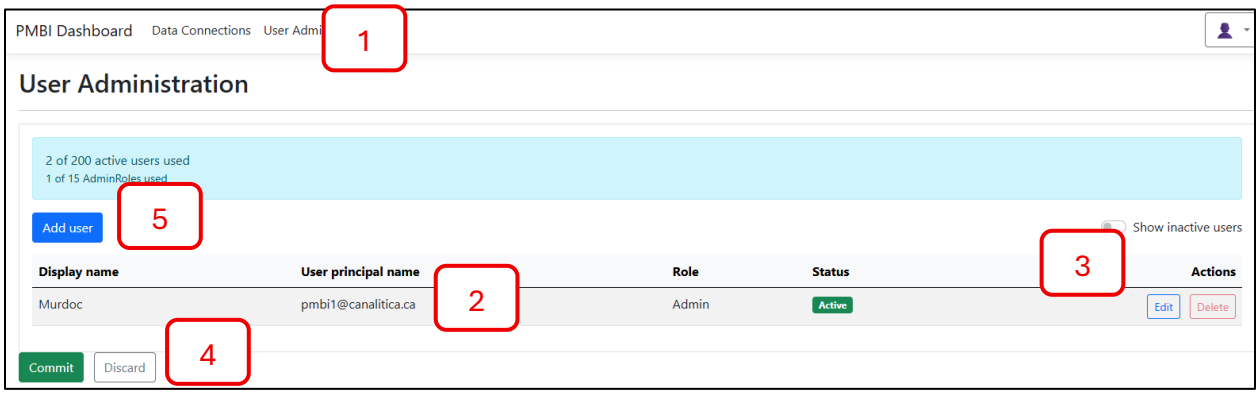


Figure 4: The User Administration page

1	Navigation bar to navigate to the main report, data connection and user administration pages.
2	List of users indicating their role and status. Inactive users are hidden by default and can be viewed using the "Show inactive users" toggle.
3	The "Show inactive users" toggle and available actions. The Edit actions allows administrators to change a user's role, project access and status. The Delete action sets a user to inactive.
4	The Commit button writes your changes to the application database. Selecting Discard will reverse any uncommitted changes.
5	The Add user button and limits associated with your plan.

Note: The number of users and administrators available in your workspace depends on your subscription plan and is displayed on screen. Refer to your plan details for current limits.

3.2 Managing Users

3.2.1 Adding a User

To add a new user to your workspace:

1. Navigate to the **User Admin** page.
2. Click **Add User** (fig. 5).

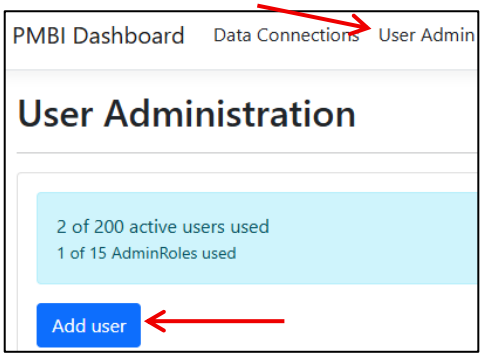


Figure 5: The Add User button

3. Enter their Microsoft Entra ID email address under **User principal name**. Note: **Display Name** will be automatically populated upon first log-in by the new user (fig. 6).
4. Select their role — **Admin** or **User**.
5. Click **Save**.

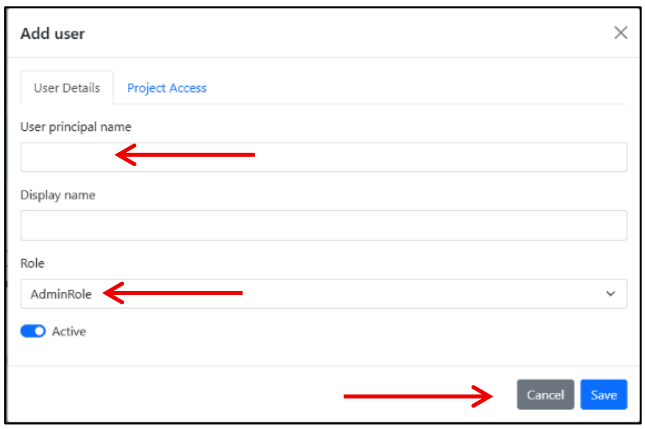


Figure 6: Entering user details

6. Finally, you must hit the **Commit** button to finalize the changes (fig. 7). Failure to do so will result in the changes not being applied. Selecting the **Discard** button will reverse your non-committed changes.

Once added, the user will be able to sign in immediately using their Microsoft account at https://pmbi.canalitica.ca. No invitation email is sent by the platform; administrators should notify new users directly.

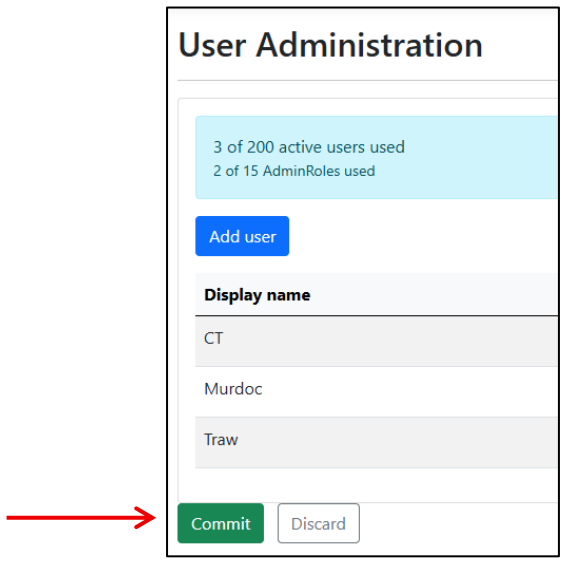


Figure 7: Commit your changes

3.2.2 Editing a User

To change a user's role or status:

1. Navigate to the **Users Admin** page (see section 3.2.1).
2. Locate the desired user and click the **Edit** button (fig. 8) in the corresponding user row. Note: The list of users is defaulted to show active users only. If an existing user cannot be found, try setting the "Show inactive users" toggle on.

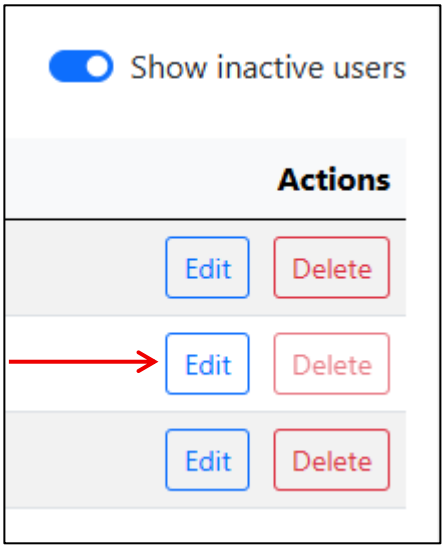


Figure 8: The User Admin Action buttons

3. Update their role or status as desired (fig. 9). Click **Save** to continue or **Cancel** to quit. Note: User Principal Name and Display Name are not editable through the app.

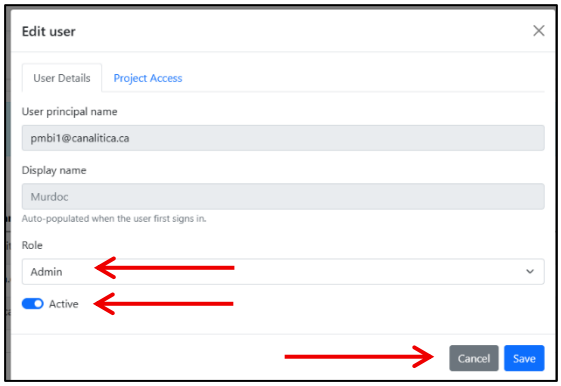


Figure 9: Updating a user's role and/or status

4. Finally, you must hit the **Commit** button to finalize the changes. Failure to do so will result in the changes not being applied. Selecting the **Discard** button will reverse your non-committed changes.

Figure 10: Always Commit your changes to take effect

3.2.3 Removing a User

Users are deactivated rather than permanently deleted, preserving any historical activity associated with their account.

1. Navigate to **Users Admin** (see section 3.2.1).

2. Locate the desired user and click the **Delete** button (fig. 11) in the corresponding user row. (you can also deactivate a user through the Edit User action)

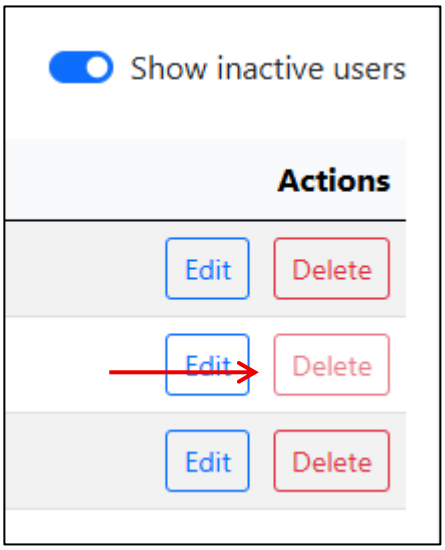


Figure 11: The Delete button sets a user to inactive

3. Finally, you must hit the **Commit** button to finalize the changes. Failure to do so will result in the changes not being applied.

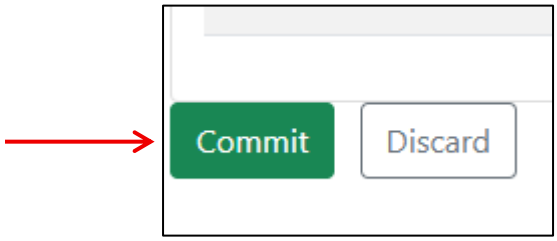


Figure 12: Commit changes

Inactive users cannot sign in. Their account remains in the system and can be reactivated at any time through the Edit action.

3.3 Managing Project Access

By default, new users are automatically granted access to all projects in your workspace. Administrators can refine this on a per-user basis at any time using the **Project Access** tab in the Edit user screen.

To adjust a user's project access:

1. Navigate to the **User Admin** page.
2. Locate the desired user and click **Edit** (see section 3.2.2).
3. Click the **Project Access** tab (fig. 13).
4. Check to grant access or uncheck to remove access to projects as appropriate.
5. Click **Save** to continue or **Cancel** to quit.

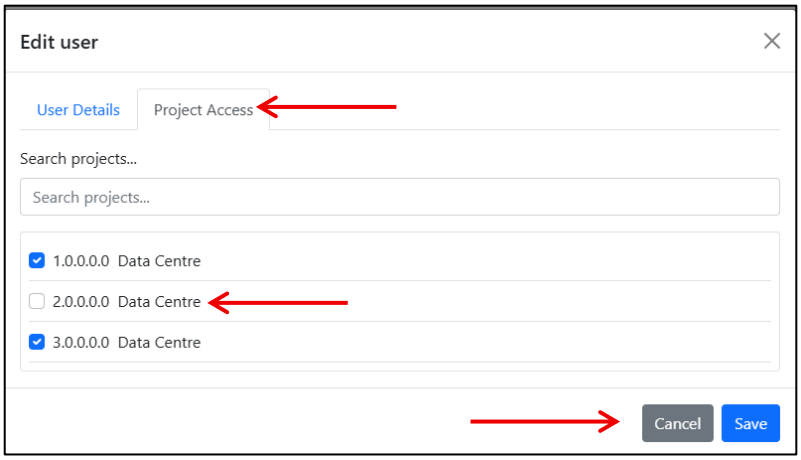


Figure 13: The Edit User window

6. Click **Commit**

Figure 14

Note: Every user must have access to at least one project. If a user's role no longer requires any project access, deactivate their account rather than removing all projects.

Note: If no projects exist in your workspace yet, the Project Access tab will indicate this. Projects are created when data is uploaded. See section 4.0 for details.

4.0 Preparing & Uploading Data

4.1 Overview

Data is loaded into CANalitica PMBI through the **Data Connections** page (fig. 15) from the main navigation. This is where you connect your project data sources, map your columns to the platform's data model, and prepare your data for reporting.

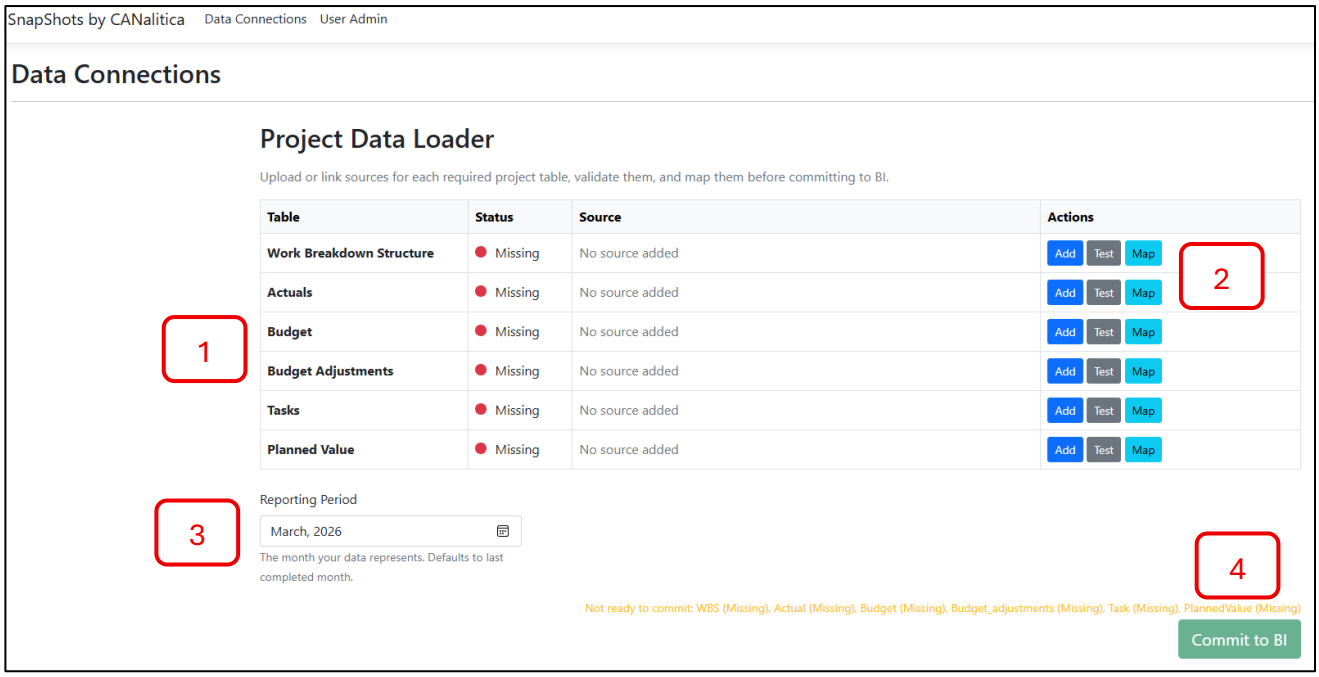


Figure 15: The Data Connection Page

1	Display of the core tables needed for PMBI, their status and source information. See sections 4.1.1, 4.1.2 and 4.1.3 for more detail.
2	List of actions. See section 4.1.4 for more detail.
3	The “Reporting Period” selection area.
4	The Commit button writes your changes to the application database.

The remainder of section 4 will be demonstrated for one core table, however, the process of adding, testing and mapping for each table is the same. Additional details will be found in the appendices.

4.1.1 Tables

The Project Data Loader requires six data tables to be added and mapped before a reporting period can be committed:

Table	Description
Work Breakdown Structure	Your project hierarchy. The WBS structure that all other tables reference.
Budget	Original approved budget values per WBS node. This is not necessarily the current budget.
Budget Adjustments	Approved changes to the original budget.
Tasks	Individual tasks.
Actuals	Costs and earned value information as of the reporting date.
Planned Value	The time-phased plan (how much value was planned to be earned over each period).

Details and structure of each table is covered in section 4.2 and in the appendices of this manual.

4.1.2 Status

Indicates the status of each table.

Status Type	Meaning
Missing	No source registered
Pending	Reserved for SQL data sources where Hybid Connection client set up is not complete yet.
Uploaded	Source exists but column mapping not yet done
Tested OK	Source connection tested but column mapping not yet done
Error	Source test failed; must be resolved before committing
Ready	Source is ready to be committed to BI
Committed	Source Data has been committed to BI

Data can only be committed when all tables are in the “Ready” state.

4.1.3 Source

Information regarding the loaded data source will be shown here. Each table can be sourced from one of the following:

Source Type	Available On
Excel (.xlsx)	All plans
CSV (.csv)	All plans
SQL Server	Pro and above plans
PostgreSQL	Coming soon for Pro and above plans
MySQL	Coming soon for Pro and above plans

Excel and CSV uploads are file-based whereas SQL Server connections are configured once and queried automatically each time you commit a period-based snapshot of the project data. Functionality for additional data sources is under development and will be added as they become available.

4.1.4 Actions

The Action buttons perform specific tasks.

Action	Meaning
Add	Used to add or edit a connection or file source for the corresponding table.
Test	Tests whether the data can be reached.
Map	Opens the Map Column screen to map source data to the target tables.

More detail can be found in section 4.2.

.

4.2 Adding a Data Source

To add or replace a source for any table:

1) On the **Data Connections** page, click **Add** next to the table you want to configure (fig. 16).

Figure 16: The Add action for a data source

2) Select your **Source Type** from the dropdown (fig. 17).
   a) For **Excel or CSV:** Click **Browse**, select your prepared file.
   b) For **SQL Server:** Enter your connection details — server name, database, and query or table name. See Section 4.2.1 for SQL connection guidance.
3) Once uploaded or connected, click **Upload** to continue or **Cancel** to discard changes and return to the Data Connections page.

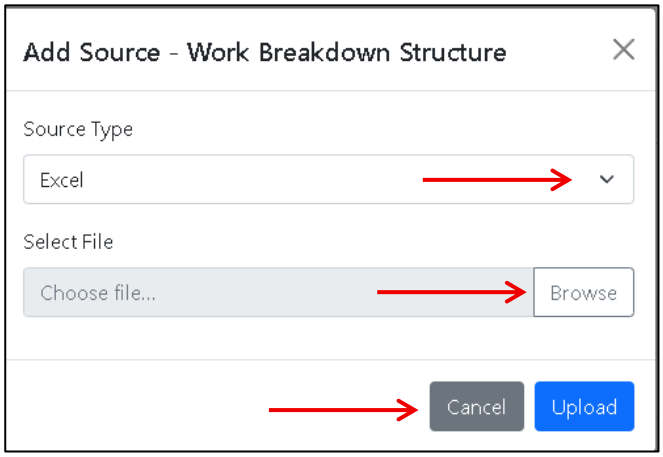


Figure 17: Adding an Excel data source

Note: Clicking **Add** on a table that already has a source will replace the existing source and reset the column mapping.

Note: The data you upload should contain the project's whole history (from inception to-date). Mostly applies to the data in the Actuals table. The Planned Value table should have planned values for the entire duration of the project.

Note: If data in an Excel/CSV file has not changed since the last reporting period (for example the WBS of the project has not changed), you do not need to re-add it. The existing data will be included in the committal.

Note: Once connection to a database, you do not need to re-connect it for subsequent reporting purposes.

4.2.1 SQL Connection Guidance

If choosing SQL Server as the source type, you will be prompted with the appropriate fields to complete (fig. 18).

The app can connect either with allowed inbound connections from the platform or you connect to your database via a Hybrid Connection Client that you can download form the Microsoft site. For Windows: click here. For Linux: click here.

- **Server:** this is your SQL Server instance name or fully qualified hostname.
- **Database:** name where your project data/tables resides.
- **Authentication:**
  - **SQL Server Authentication:** enter the SQL username and password.
  - **Windows Authentication:** uses the credentials of the service account running the platform; no username or password required.
- **Source Table Name:** the table or view in your database that contains the relevant project data associated with the corresponding PMBI core table.
- **Test Connection** to verify the platform can reach your database and read the specified table before saving.

Once connected and tested, click **Connect** to continue or **Cancel** to discard changes and return to the Data Connections page.

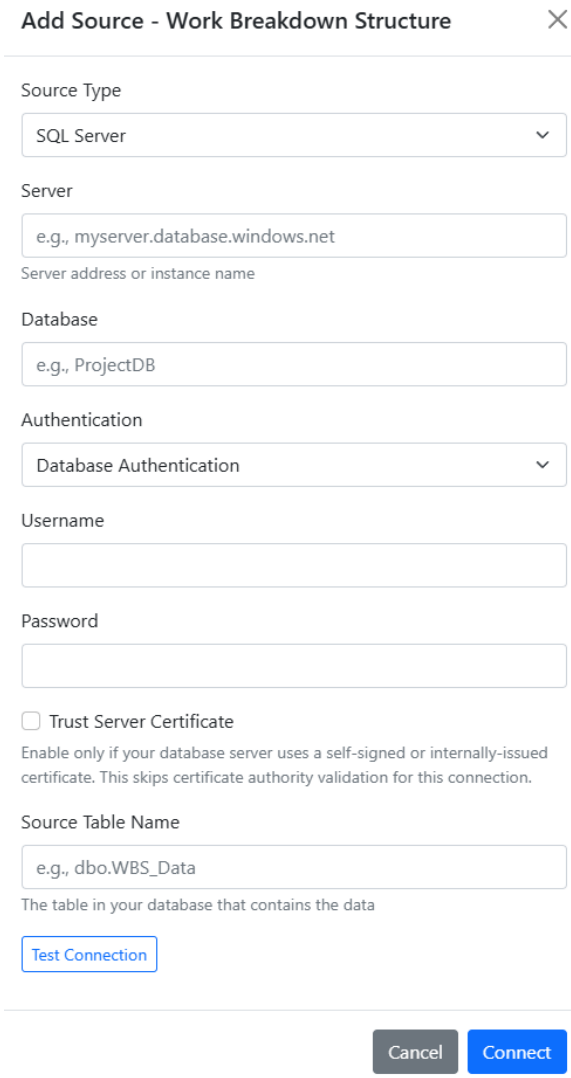


Figure 18: Adding a SQL Server data source

When connecting with a database for the first time, the status will change to “Pending” while CANalitica sets up a Hybrid Connection for your organization. Once that is completed, the dedicated connection string will appear in your Data Connection page. Use this string to complete the connection using the Hybrid Connection GUI provided by Windows.

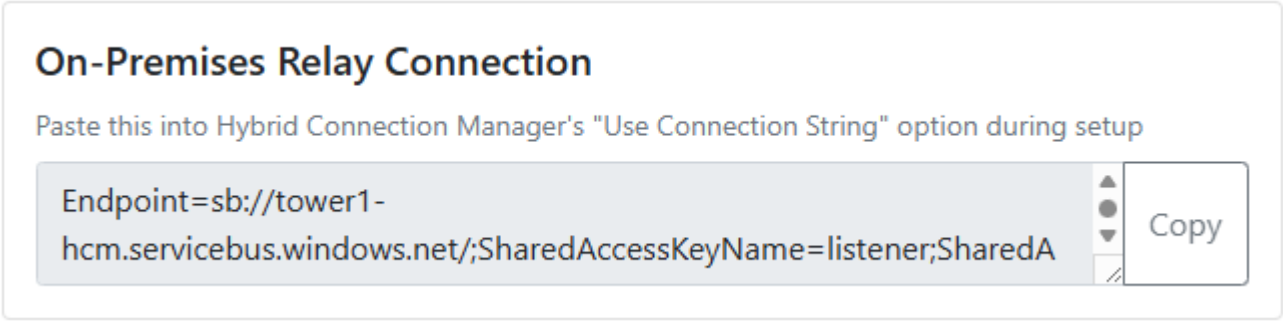


Figure 19: Once CANalitica completes your Hybrid Connection setup, you will be provided a connection string within the Data Connection Page.

Alternatively, if your IT prefers to use outbound IP addresses, you may set that up prior to attempting to connect with your database. Contact pmbisupport@canalitica.ca for the current list of outbound IP addresses or refer to the resource page on our website (https://canalitica.ca/resources) to whitelist on your firewall or Azure SQL firewall rules.

Tip: Using database views rather than raw tables as your source gives you flexibility to shape, filter, and join your data in the view without changing your underlying schema. This is the recommended approach for organizations with complex project databases and should be discussed with your database administrators.

Note: Your SQL Server must be accessible from the CANalitica PMBI platform. If your database is behind a firewall or private network, your network administrator will need either use a Hybrid Connection client or allow inbound connections from the platform. Contact pmbisupport@canalitica.ca for the current list of outbound IP addresses or refer to the resource page on our website (https://canalitica.ca/resources) to whitelist on your firewall or Azure SQL firewall rules.

Security note: Your credentials are encrypted and stored securely and used only to retrieve data during uploads and creating reporting snapshots. CANalitica recommends creating a dedicated read-only database user for this connection rather than using an administrative account.

4.3 Testing a Data Source

Selecting the **Test** action will attempt to reach the data source and inform the user with a message and change in Status (fig. 19 and 20).

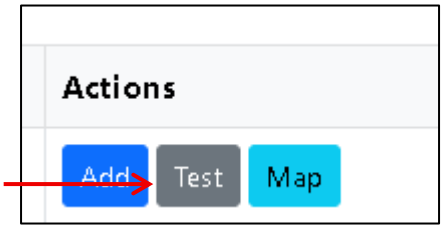


Figure 20: Data source Test action button

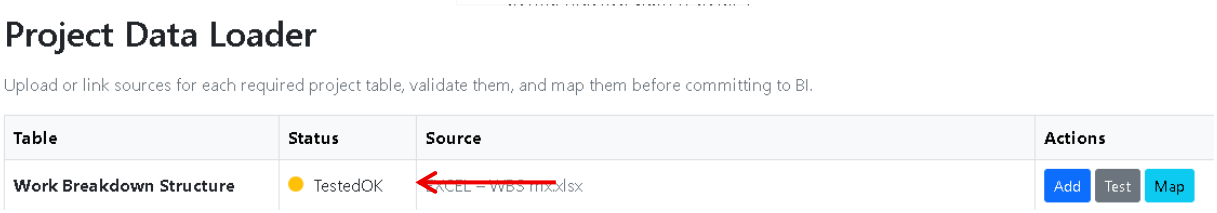


Figure 21: PMBI core table in the Tested OK status

4.4 Mapping a Data Source

Column mapping is how you tell CANalitica PMBI which column in your source data corresponds to which field in the reporting model. Clicking on the **Map** button (fig. 21) will open the Map Source Columns dialog (fig. 22) and allow the user to select the appropriate PMBI column.

The platform suggests mappings based on column name similarity. You can use column naming conventions of your choice, but if you use one similar to the ones provided in the appendices, it will make mapping columns faster. Full column definitions and expected names for each table are provided in the Appendices.

To review and confirm mappings:

1. Click **Map** next to the relevant table on the Project Data Loader page (fig. 19).
2. Review each row and confirm the source column is mapped to the correct model column (fig. 20).
3. If a mapping is incorrect, use the dropdown to select the correct model column.
4. Click **Save Mappings** when all rows are confirmed or Cancel to exit.

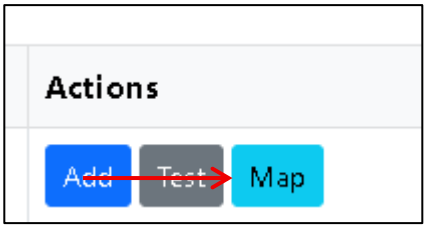


Figure 22: The core table Map action button

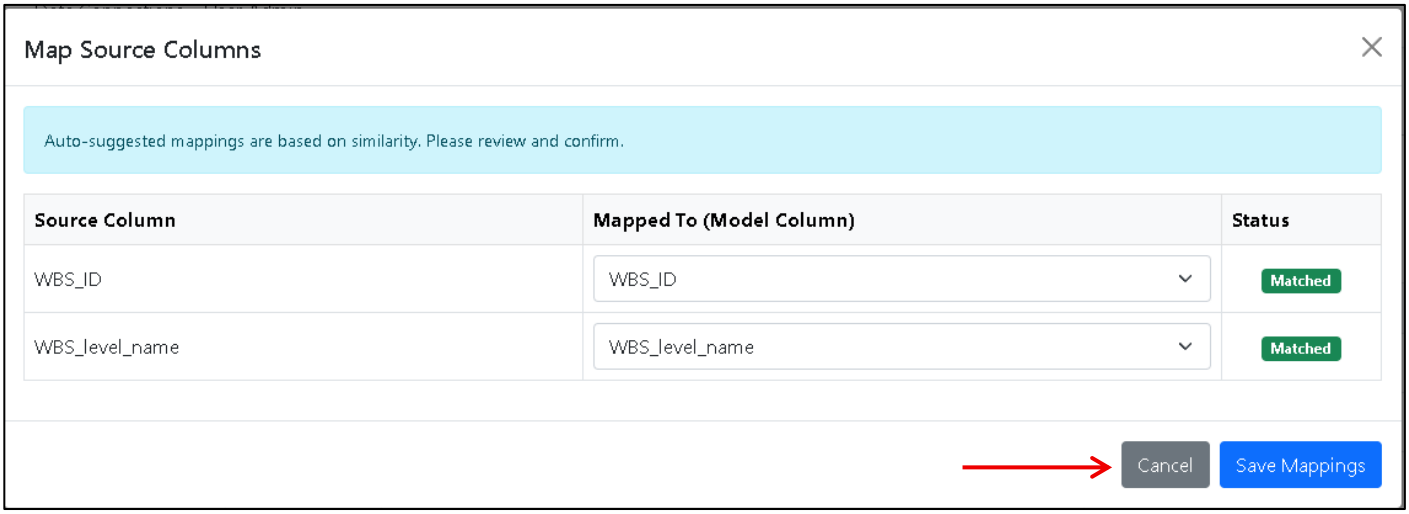


Figure 23: The Map Source Column screen

Note: The Source Column will display the columns in your data file/tables. You are required to map those columns to PMBI's internal database columns indicated by the "Mapped To (Model Column" fields.

Once mapped, the status to the table will change to “Ready” (fig. 23). Continue the steps from sections 4.2 through 4.4 for the remaining PMBI core tables.

Note: If the data for any of the PMBI tables hasn’t changed since the last upload, you do not need to re-add or re-map that table. It’s connection and mapping will be retained and the data will be re-committed (see section 4.5).

Table	Status
Work Breakdown Structure	Ready
Actuals	Ready
Budget	Ready
Budget Adjustments	Ready
Tasks	Ready
Planned Value	Ready

Figure 24: PMBI core tables with the Ready status

4.5 Committing the Data

Once all PMBI tables have the “Ready” status, you are ready to commit the data to the dashboard. Note: The number of Commits available in your workspace depends on your subscription plan and is displayed on screen. Refer to your plan details for current limits

1) Select the **Reporting Period** (month-year) that you are committing.
2) Select the **Commit to BI** button

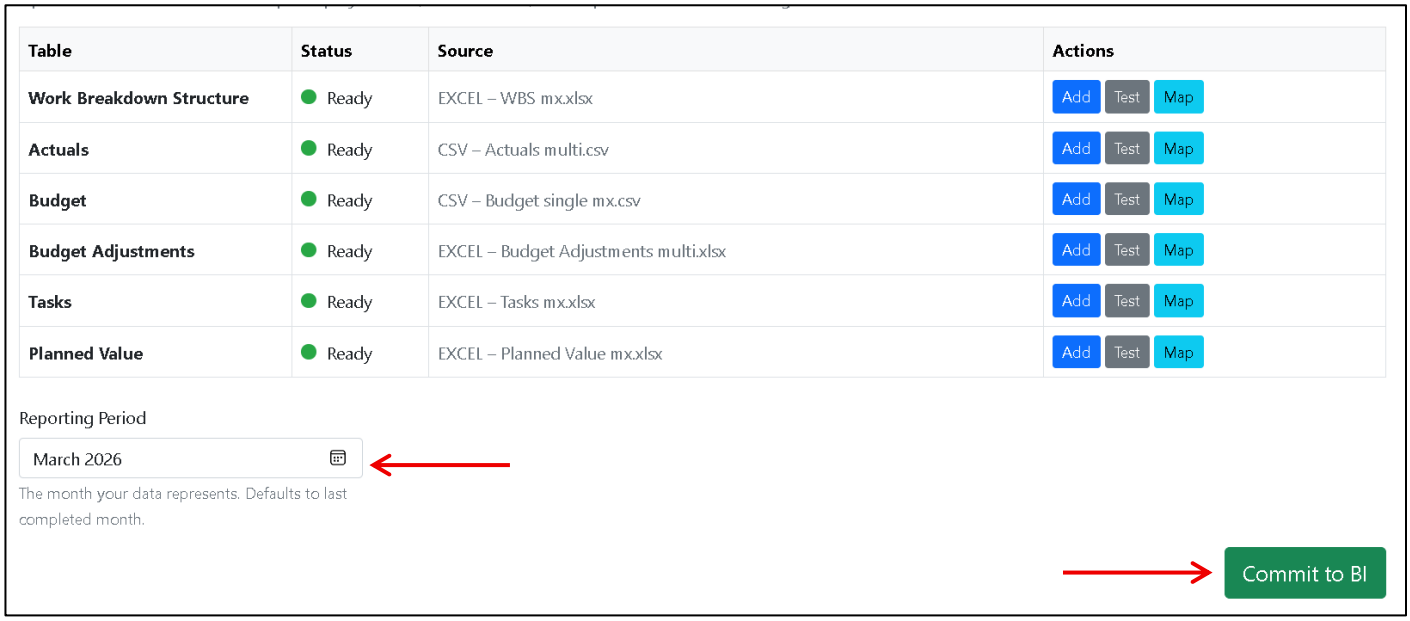


Figure 25: Selecting the reporting period ahead of committing to BI

3) If your data has multiple projects, you can select all or some for commitment on the preview window (fig. 25). Unselected projects will not be part of the that commit process.
4) Select **Confirm & Commit** to contine to **Cancel** to return to the Data Connections page.

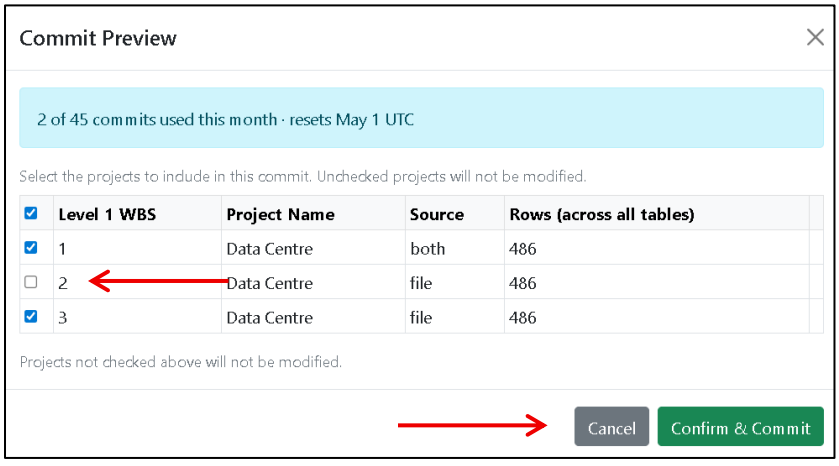


Figure 26: Project selection on the Commit Preview window

4.5.1 Commit Messages and Notifications

During the committal process, the application will perform various validation checks and you may be prompted with circumstantial messages. Below are the various messages you may receive and their meaning:

- **Snapshot Already Exists:** If you commit data for a period that was previously committed, you will be warned with a prompt (fig. 26) and be asked if you want to create a revised snapshot for the period. This action will overwrite the dashboard data for that period with the newer snapshot. Cases where this may be desired are if new or missing information was realized after the original committal and therefore the original snapshot became outdated/inaccurate.

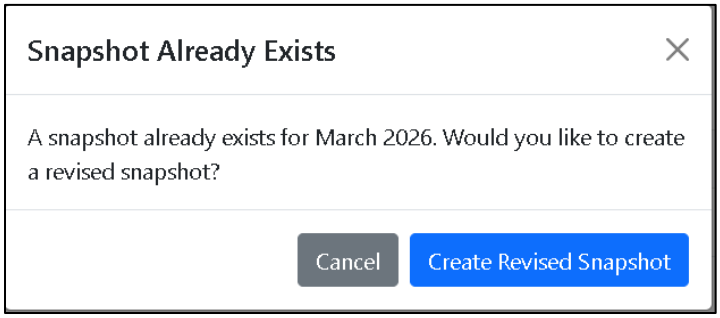


Figure 27: Snapshot exists message

- **Regenerate Subsequent Snapshots:** This prompt (fig. 27) will appear when a historical reporting period is being revised, thereby impacting all subsequent cumulative values of future reporting periods and possibly rendering these periods stale. In most cases, it is recommended to select Regenerate Subsequent Snapshots so that reporting periods after the revised period will include the updated information. Cases where you would want to maintain existing snapshots could be where data errors are known in other periods and you wish to revise them on a month-by-month basis.

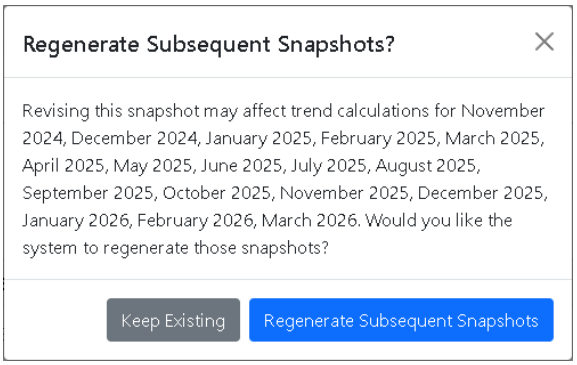


Figure 28: Regenerating subsequent reporting periods.

- **Missing Snapshots Detected:** When there are gaps in reporting periods, the system will alert you and give you the option to auto-generated snapshots (fig. 28). The application will not allow gaps in reporting periods. If you choose not to auto-generate, the commit process will stop. You have the option to then manually create snapshots for reporting period in succession.

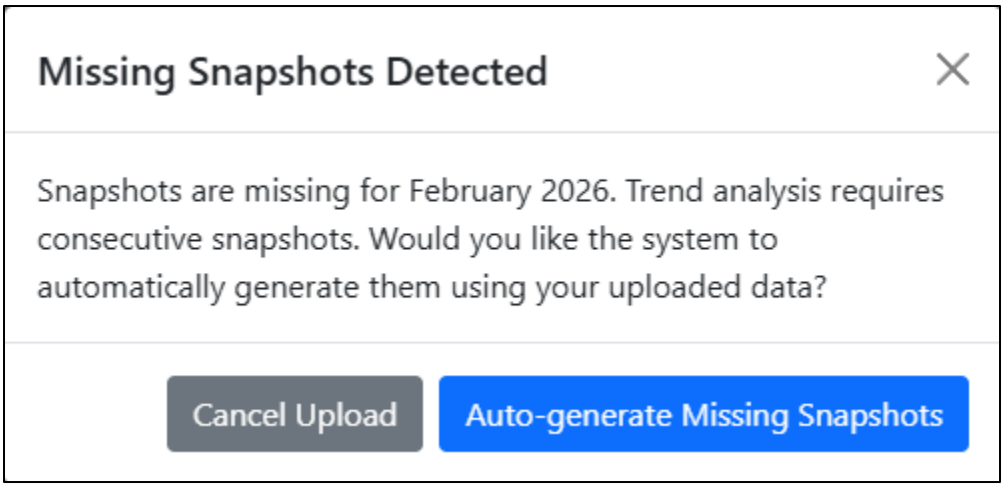


Figure 29:Missing Snapshots Detected prompt

- **Commit Summary:** Once all the validation is done, you will arrive at the Commit Summary window (fig. 29 and 30). At this window, you may receive messages describing issues with an unsuccessful commit or a summary of a successful commit. Unsuccessful commits do not count toward commit limits and result in no data being written to the database.

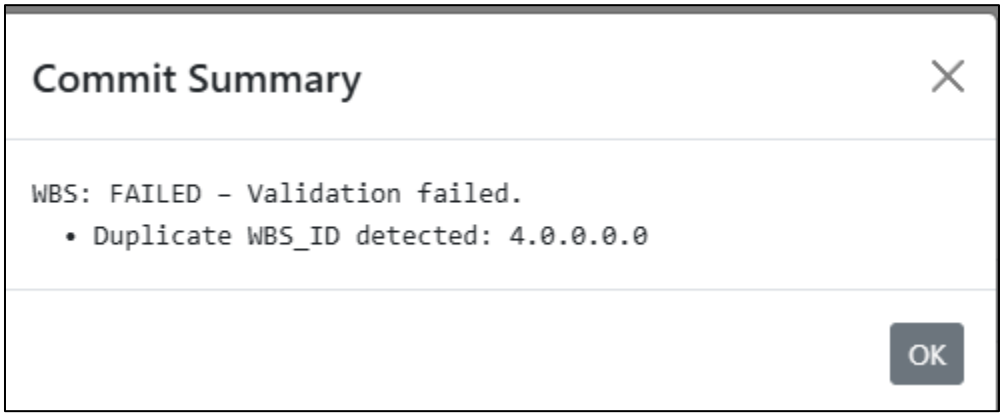


Figure 30: An unsuccessful commit due to a duplication error.

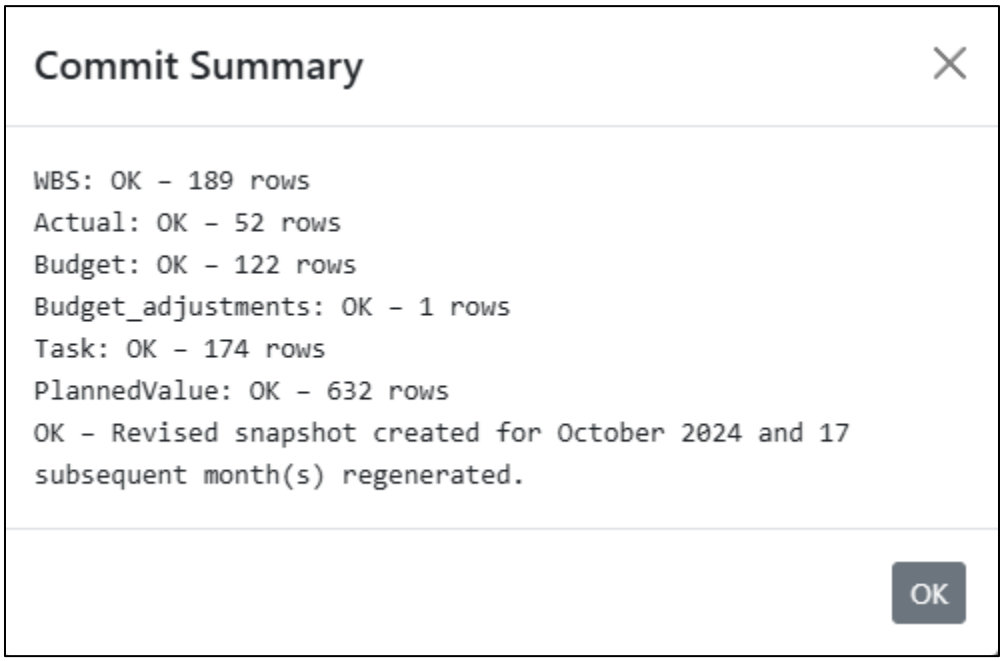


Figure 31: A successful commit summary indicating rows that were written.

4.5.2 Data Refresh in BI

Once committed, the PMBI dashboard will be refreshed. A windowed message will appear indicating the status of the refresh. Depending on the subscription plan, the refresh can occur in near real-time or at the top of the next hour.

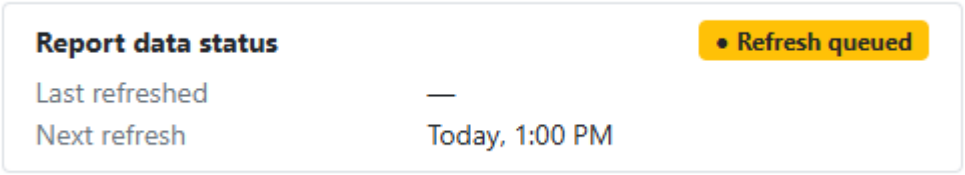


Figure 32: PMBI Refresh Status Window

5.0 Main Dashboard

Upon signing-in, you will be directed to the main dashboard page (fig. 31). You can also reach this page by selecting the PMBI Dashboard link in the navigation bar.

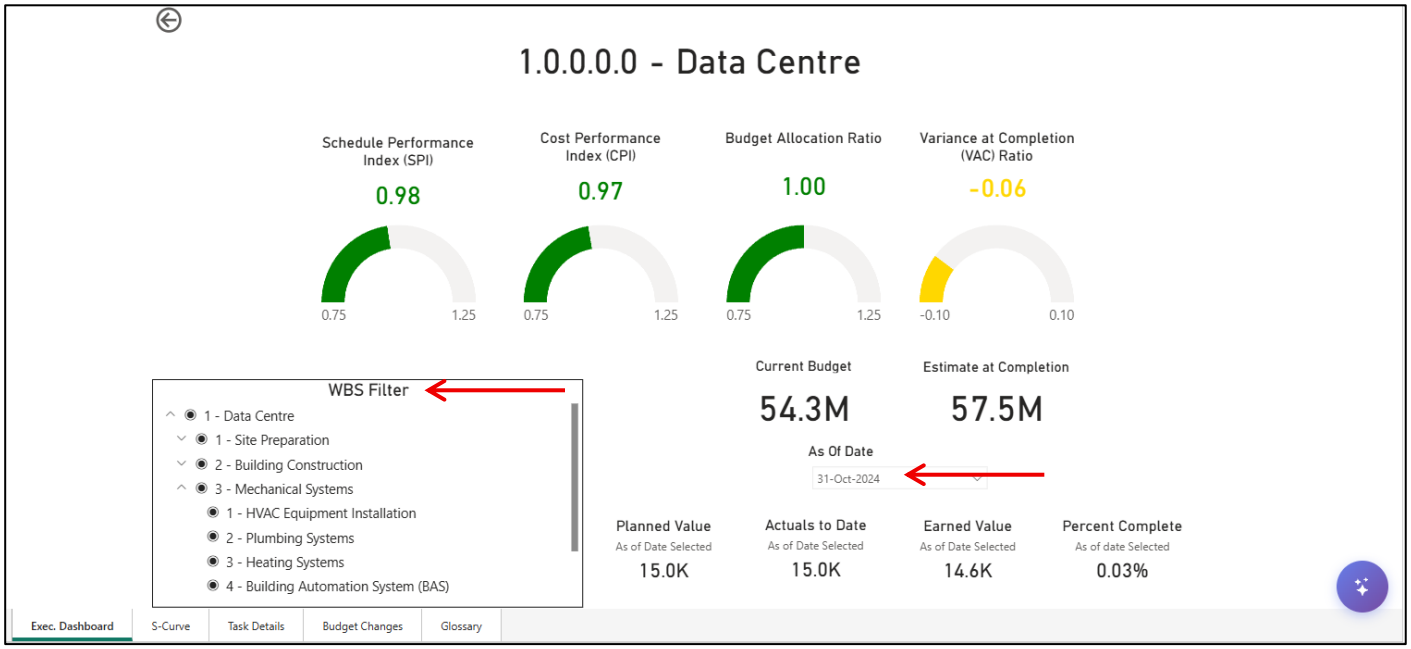


Figure 33: The main dashboard page with the two common filters pointed out.

5.1 Navigating the Dashboard

5.1.1 Report Pages

If you are familiar with Microsoft Power BI or Excel, you will find navigation straightforward. Simply select the desired page along the bottom bar (fig. 32).

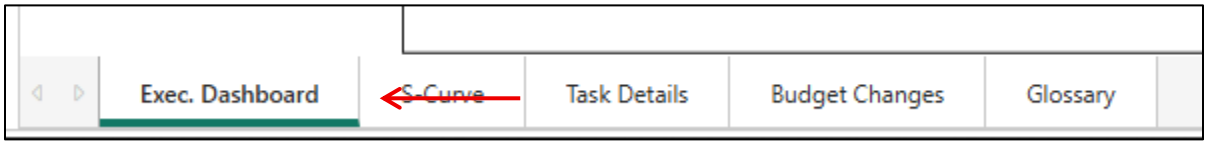


Figure 34: The dashboard page navigation bar.

5.1.2 Slicers and Filters

There are 2 slicers (or filters) that are common to all pages: the WBS filter (fig. 33) and As Of Date (fig. 34) filter. These slicers are synched and changing a selection on one page will be mirrored on the other dashboard pages.

Filters that are specific or work with certain pages or visuals will be discussed in the sections related to those pages or visuals.

WBS Filter

This filter allows you to focus on one WBS item including its children elements. The WBS filter only goes down 3 out of the 5 available WBS levels for ease of use.

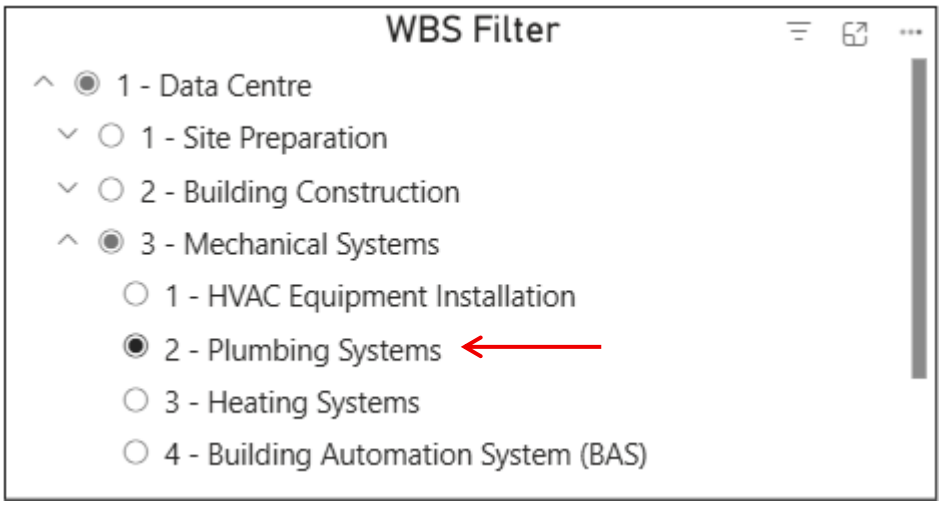


Figure 35: The WBS filter.

As Of Date Filter

The "As Of Date" filter serve an important function of displaying project performance data for that reporting period. For most cases, management would be interested in the most recent reporting period. The filter will present the reporting periods that have been committed from the data loader page. If no reporting period exists in the filter, it must first be committed to the dashboard (see section 4.5).

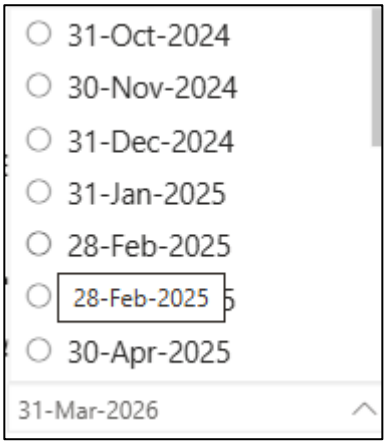


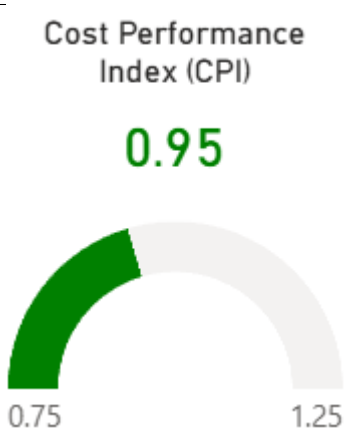
Figure 36: The As Of Date filter and the selection list.

5.2 Understanding and Using the Dashboard

This section will describe each page, the visuals within them and explain how to use the report to improve your organization's project reporting to enable better project delivery.

5.2.1 Exec. Dashboard Page

The following visuals are found on the Executive Dashboard page.

	**Schedule Performance Index (SPI)** Measures how efficiently the project is progressing against its planned schedule. A value of **1.00 means on schedule** (above 1.00 means ahead, below 1.00 means behind). The gauge shown reads **0.87**, indicating the project is delivering work more slowly than planned (roughly 87 cents of scheduled work completed for every dollar of work that should have been done by now).
	**Cost Performance Index (CPI)** Measures how efficiently the project is spending its budget relative to the work accomplished. A value of **1.00 means on budget** (above 1.00 means under budget, below 1.00 means over budget). The gauge shown reads **0.95**, indicating the project is slightly over budget (approximately 95 cents of work completed for every dollar spent).
	**Budget Allocation Ratio (BAR)** Measures how much of the project's total authorized budget has been assigned or "planned" to specific tasks by way of planned value. A value of **1.00 means 100% of the budget is allocated to tasks (or PV = Current Budget)**. A value **below 1.00** means a portion of the budget has not yet been planned to any task, which could indicate scope that hasn't been planned yet, or an intentional reserve being held outside the task structure. A value **above 1.00** would indicate tasks have been budgeted for more than the total authorized amount, which would warrant investigation.

Variance at Completion (VAC) Ratio -0.06  	**Variance at Completion (VAC) Ratio** Measures the projected budget surplus or shortfall at project completion, expressed as a ratio between the Estimate at Completion relative to the current authorized budget. A value of **0.00 means the project is forecast to be finished exactly on budget**. A **positive value** indicates a projected surplus; a **negative value** indicates a projected overrun. The gauge shown reads **-0.06**, meaning the project is currently forecast to finish approximately 6% over its authorized budget.
Current Budget 54.3M	**Current Budget** The total authorized budget for the project, reflecting the original baseline plus any approved changes.
Estimate at Completion 57.5M	**Estimate at Completion (EAC)** The current forecast of the total cost to complete the entire project, based on actual spending to date and the expected cost of remaining work.
Planned Value As of Date Selected 4.0M	**Planned Value (PV) — As of Date Selected** The budgeted value of all work that was *scheduled* to be completed by the selected reporting date. This represents what *should* have been accomplished in dollar terms according to the original plan
Actuals to Date As of Date Selected 3.6M	**Actuals to Date — As of Date Selected** The actual costs incurred for work performed up to the selected reporting date.
Earned Value As of Date Selected 3.5M	**Earned Value (EV) — As of Date Selected** The budgeted value of the work completed by the selected reporting date, regardless of what it cost. See the appendix for an in-depth look at Earned Value.
Percent Complete As of date Selected 6.41%	**Percent Complete — As of Date Selected** The proportion of the total project scope completed by the selected reporting date, calculated as cumulative Earned Value divided by the total Planned Value at completion.

5.2.2 S-Curve Page

S-Curve — Cumulative EVM Performance Chart

This chart (fig. 35) plots three cumulative cost lines over the project timeline, allowing you to visually track schedule and cost performance at a glance. The horizontal axis shows the reporting period end date; the vertical axis shows cumulative value in dollars. The characteristic S-shape of the curve reflects typical project spending patterns: slow ramp-up early, accelerating through the middle, and tapering toward completion. The three lines are:

- **Planned Value (PV) — orange:** The cumulative budget baseline — what was scheduled to be accomplished by each point in time according to the project plan.
- **Earned Value (EV) — dark blue:** The cumulative budgeted value of work actually completed. When the EV line is below the PV line, the project is behind schedule.
- **Actual Costs (AC) — light blue:** The cumulative real costs incurred to date. When the AC line is above the EV line, the project is spending more than the value it is producing (cost overrun).

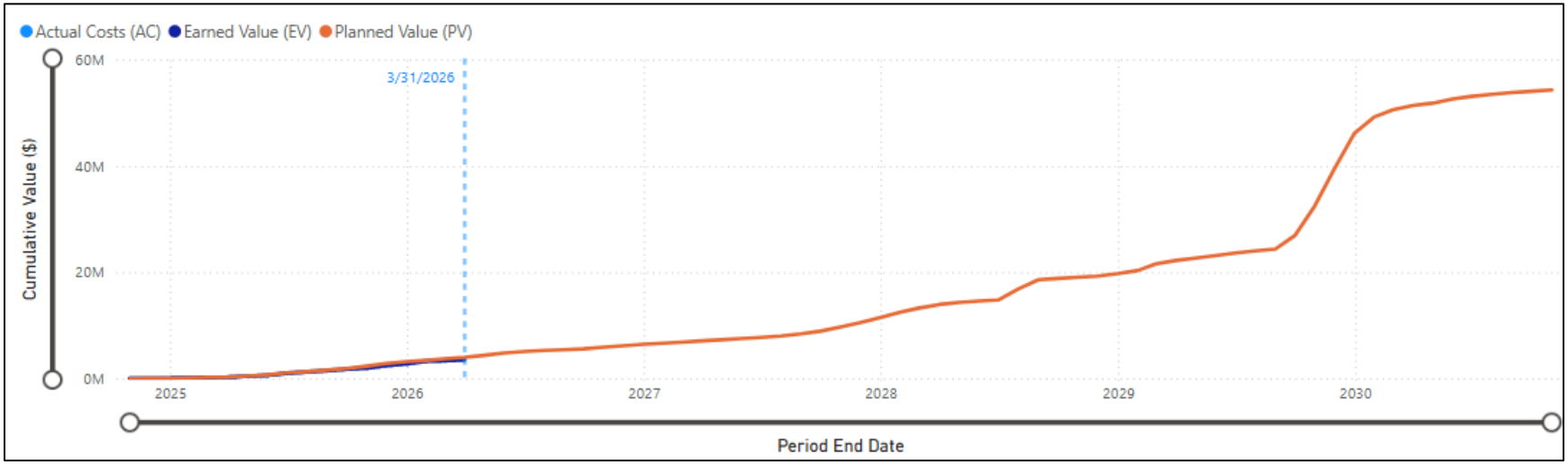


Figure 37: The PMBI S-Curve

The chart shown can be zoomed in using the period slider controls on both axes (fig. 36). In the below figure, the focus was adjusted between to better see the variances. The tooltip at the selected date shows the recorded values.

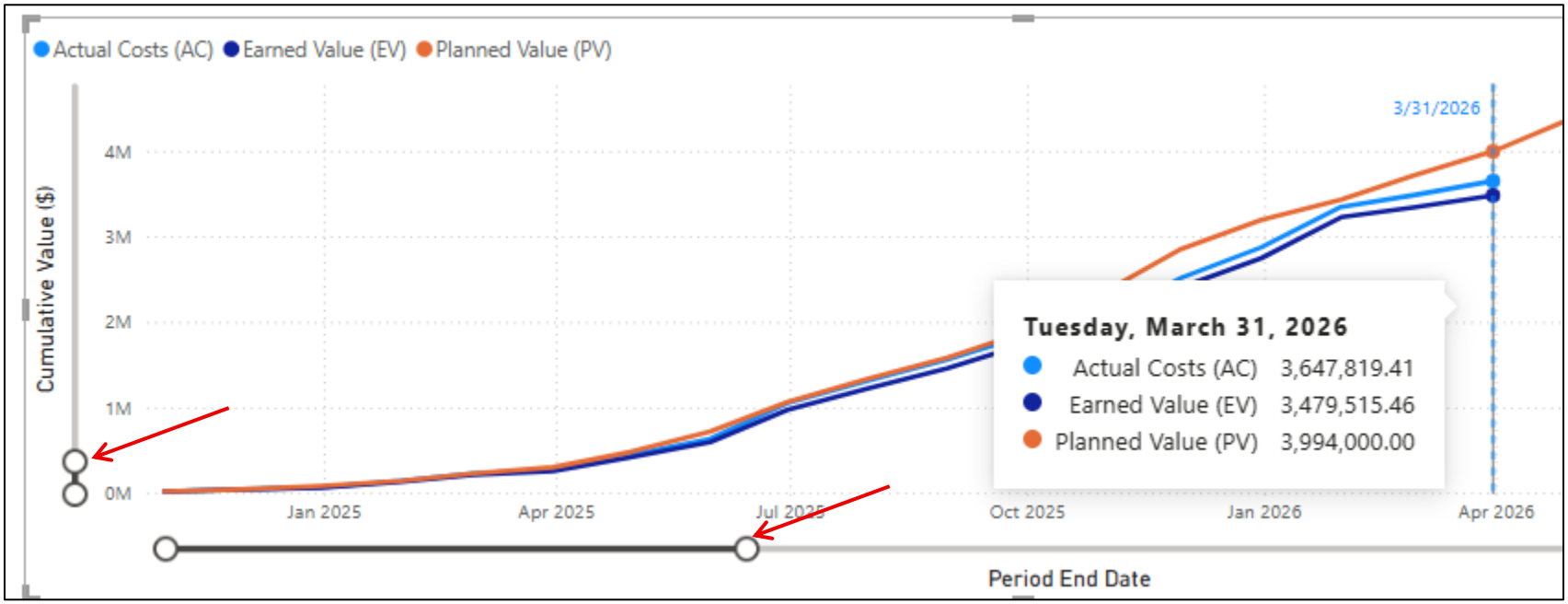


Figure 38: Using sliders on the S-Curve

CANalitica

5.2.3 Task Details Page

Task Details Table

The Task Details (fig. 35) table provides a row-by-row breakdown of Earned Value Management (EVM) metrics for every active work package in the project. Where the summary gauges and S-Curve give you a project-level health signal, this table lets you drill into the performance of individual tasks to identify which specific areas are driving schedule delays, cost overruns, or both.

Columns at a Glance

Each row represents a single task or work package and displays the following metrics:

Column	What It Shows
Task	The name of the work package
EAC	Estimate at Completion — the current forecast of total cost for this task
Total PV	The full planned value (budget) for this task over its entire duration
PV to Date	The portion of the task's budget that was scheduled to be complete as of the selected reporting date
Costs to Date	Actual costs incurred on this task as of the selected reporting date
EV to Date	The budgeted value of work actually completed on this task as of the selected reporting date
SPI	Schedule Performance Index for this task — below 1.00 means behind schedule
CPI	Cost Performance Index for this task — below 1.00 means over budget for work done
% Complete	Budget-weighted progress for this task, calculated as cumulative EV divided by Total PV

The **Total row** at the bottom summarizes all tasks and should reconcile with the project-level metrics shown in the summary gauges and S-Curve.

Filtering and Sorting

The table supports interactive filtering and sorting to help you focus on the tasks that matter most:

- **Sort any column** by clicking its header. The small arrow indicator on the **PV to Date** column shown in figure 37 indicates it is currently sorted. Clicking again reverses the sort order. This is useful for quickly ranking tasks by cost, schedule performance, or completion percentage.
- **Filter by reporting date** using the date selector on the page — all "to date" columns update automatically to reflect the selected period, allowing you to review performance at any historical point in time.
- **Cross-filter with other visuals** on the page — selecting a task row will highlight that task's contribution in the S-Curve and other visuals, and vice versa.

How to Use This Table

A few practical ways to get value from this view:

- **Spot underperforming tasks early** by sorting on SPI or CPI ascending — tasks at the top of the list are your most at-risk work packages.
- **Identify your major tasks by scale** by sorting on Total PV, EAC, Costs to Date, or EV to Date in descending order — the tasks that rise to the top represent the largest planned commitments, biggest cost exposures, or most work accomplished. This is useful for focusing management attention where budget and effort are most concentrated, and for understanding which tasks will have the greatest impact on overall project performance.
- **Compare EAC to Total PV** to see which tasks are forecast to overrun their individual budgets before the project is finished.

Task	EAC	Total PV	PV to date	Costs to date	EV to date	SPI	CPI	% Complete
Permitting	1,219,469	1,215,000	1,215,000	1,046,169	1,024,987	0.84	0.98	84.36%
Ongoing Project Management	8,313,657	7,996,450	1,149,000	1,033,527	940,971	0.82	0.91	11.77%
Final Design	478,587	425,000	425,000	404,835	401,890	0.95	0.99	94.56%
Utility Agreements	394,717	365,000	365,000	339,086	338,541	0.93	1.00	92.75%
Detailed Design	319,969	315,000	315,000	308,215	280,652	0.89	0.91	89.10%
Prelim Design	303,965	289,000	289,000	288,867	266,475	0.92	0.92	92.21%
Design Approval	111,824	100,000	100,000	92,530	97,358	0.97	1.05	97.36%
Ongoing Legal Advice	1,286,925	1,190,600	87,500	87,059	83,804	0.96	0.96	7.04%
Requirement Gathering	52,572	48,500	48,500	47,531	44,837	0.92	0.94	92.45%
Deficiency Resolution	1,316,431	1,248,000	0	0	0			0.00%
Substantial Completion	28,129	25,000	0	0	0			0.00%
Total	**13,826,245**	**13,217,550**	**3,994,000**	**3,647,819**	**3,479,515**	**0.87**	**0.95**	**26.32%**

Figure 39: The Task Details Table with the "PV to date" column sorted

5.2.4 Budget Changes Page

The Budget Changes page (fig. 38) provides a transparent record of how the project's authorized budget has evolved from its original baseline. Rather than showing a single budget figure, this page breaks the budget into its two components; the original approved amount and any subsequent adjustments. This allows stakeholders to clearly see how much of the current budget was planned from the outset versus added or removed over the course of the project.

Each row represents a WBS element and displays three columns:

Column	What It Shows
Original Budget (A)	The budget established at project baseline for this WBS element
Budget Adjustments (B)	Any approved increases or decreases made to the budget after the original baseline was set
Approved Budget (A + B)	The current total authorized budget for this WBS element, reflecting all changes to date

The **Total row** at the bottom summarizes across all WBS elements and represents the project's current authorized budget in full — the same figure shown as **Current Budget** in the summary gauges.

Why This Page Is Useful

- **Accountability and audit trail:** Separating original budget from adjustments makes it immediately clear whether a project's cost growth is due to scope changes that were formally approved, rather than simply poor cost control.
- **Scope change visibility:** Large values in the Budget Adjustments column are a signal that the project scope has changed significantly since baseline, which may warrant a conversation about schedule and delivery expectations as well.
- **Reconciliation:** The Approved Budget total on this page should always reconcile with the Current Budget figure shown on the summary page, providing a built-in cross-check for data integrity.

WBS ID	WBS Name	Original Budget (A)	Budget Adjustments (B)	Approved Budget (A + B)
5.6.3.0.0	Project Management	6,996,450	1,000,000	7,996,450
Total		**6,996,450**	**1,000,000**	**7,996,450**

Figure 40: The Budget Changes (or Adjustments) table

5.2.5 AI Summary

The AI Summary (fig. 39) is an optional side-bar panel that generates a written narrative describing the current health of your project based on the EVM data loaded into PMBI. It is accessed by clicking the AI summary button on the reporting page and can be opened or dismissed at any time without affecting the underlying report data.

How It Works

When you request a narrative, PMBI assembles a structured summary of your project's key EVM metrics and submits this summarized data to the selected AI model. The model then produces a plain-language description of what the numbers indicate about project health.

It is important to understand what this feature is and is not intended to do. The AI narrative is designed to summarize and describe the project data already present. It does not perform independent analysis, derive new insights, or access any information beyond what has been submitted to it.

Because the narrative is generated by an AI model, it can occasionally misinterpret results particularly in cases such as early-stage projects with limited data, projects with unusual cost or schedule profiles, or situations where the underlying data contains anomalies. **Always review the narrative critically and cross-reference it against the report visuals.** The narrative is a communication aid, not an authoritative assessment.

A disclaimer is displayed beneath the model selector as a reminder: *AI can make mistakes. Please double check responses.*

Panel Controls

AI Model The dropdown at the top of the panel allows you to select which AI model will generate the narrative. The model shown in the example below is claude-sonnet-4-6. The available models may vary depending on available models offered. Different models may produce slightly different narrative styles or levels of detail, but all are working from the same underlying project data snapshot.

Project Selects the project for which the narrative will be generated. If your account has access to multiple projects, use this dropdown to switch between them. The narrative will always reflect the data for the project selected here, independent of any filters or selections active in the main report at the time.

Reporting Period Selects the period end date for the narrative. The AI model will describe project health as of that period. This allows you to generate narratives for historical periods, not just the most recent reporting date. **Tip –** Ensure the same reporting period is selected on the PMBI dashboard so that the AI narrative aligns with the report visuals.

Health Status and Confidence Badges

Directly below the panel controls, two badges summarize the deterministic health assessment derived from your project's EVM metrics:

The **health status badge** (Green, Amber, or Red) reflects the project's overall performance status as of the selected period, based on the Cost Performance Index and Schedule Performance Index values. This assessment is based on the data directly and is not generated by the AI model. It serves as an at-a-glance signal before you read the narrative.

The **confidence badge** indicates how much data was available to support the narrative and health assessment. A High Confidence rating means the model had sufficient EVM activity across the project to draw on. Lower confidence ratings indicate that the project may be in its early stages, have limited earned value recorded, or have sparse activity in the selected period.

Narrative Tabs

The narrative is organized into four tabs, each focusing on a different dimension of project performance:

Schedule Describes how the project is performing against its planned timeline.

Cost Describes how efficiently the project is converting spending into completed work.

Key Risk Drivers Highlights the work packages or project conditions that represent the most significant risks to schedule and cost performance.

Confidence Explains the basis for the confidence rating shown in the badge above the narrative.

Refreshing the Narrative

The **Refresh** button at the bottom of the panel regenerates the narrative by resubmitting the current project data snapshot to the AI model. Use this if you have changed the project or reporting period selection. The timestamp shown above the Refresh button indicates when the currently displayed narrative was last generated.

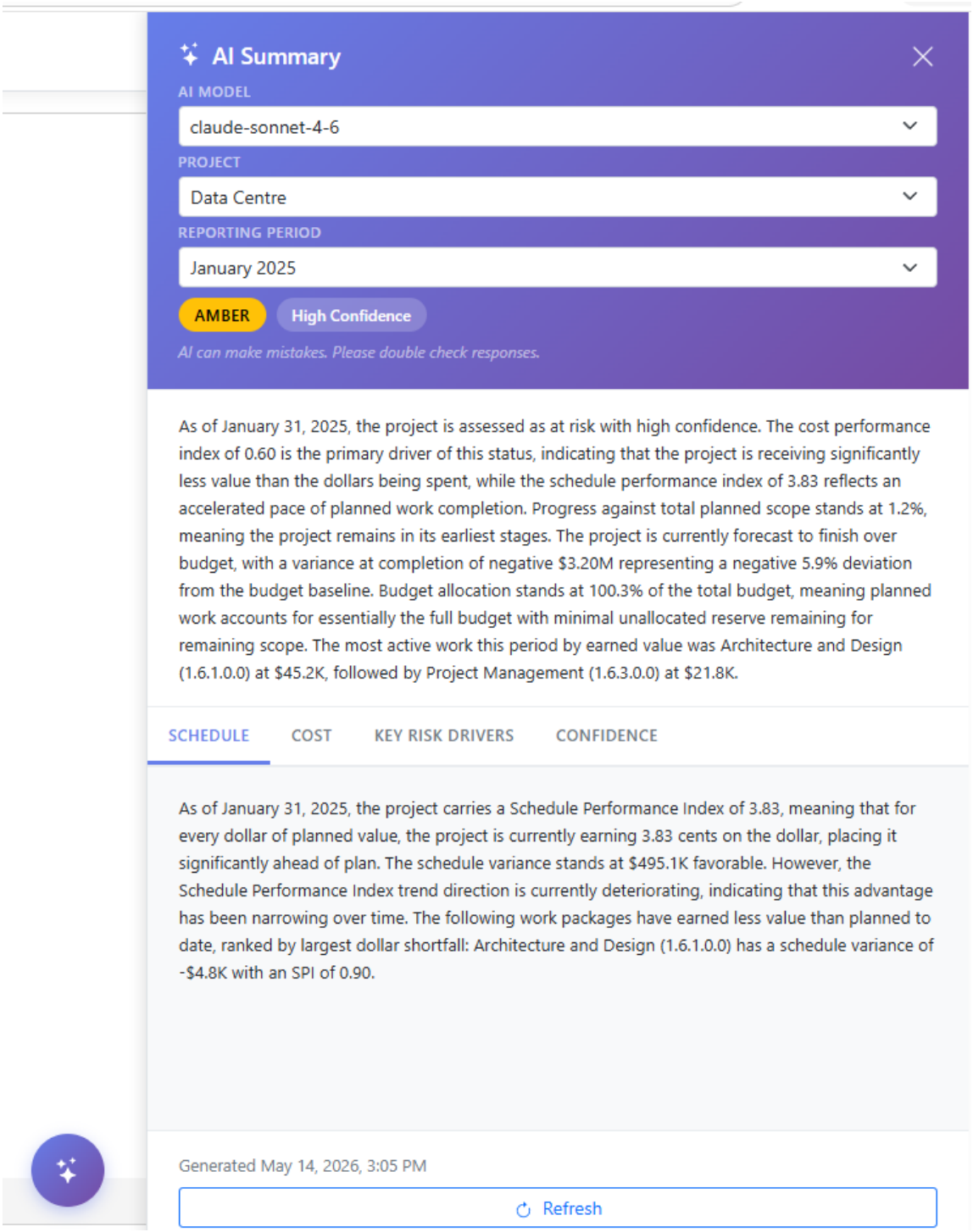


Figure 41: The AI Summary Side-Bar

Appendicies

A1 – PMBI Tables

A1.1 Work Breakdown Structure (WBS) Table

WBS_ID	WBS_level_name
[nvarchar]	[nvarchar]

Figure 42: The WBS table columns and data type

What is a WBS?

A Work Breakdown Structure (WBS) is a hierarchical decomposition of the total scope of work for a project. It organizes the project into progressively smaller, more manageable components; from the highest-level project summary down through phases or major deliverables, and ultimately to individual work packages where cost and schedule are planned and tracked.

The WBS represents the *what* of the project (the deliverables and outcomes to be produced, not the *how* or *when*). Each level of the hierarchy breaks the level above it into its component parts, and the sum of all child elements equals 100% of the scope of the parent above them.

Every work package in the WBS becomes the foundation for budgeting, scheduling, resource assignment, and performance measurement. In CANalitica PMBI, the WBS is the backbone of all EVM reporting: cost performance, schedule performance, earned value, and percent complete are all calculated and rolled up through the WBS hierarchy.

The Root Node (or Project Root)

WBS_ID	WBS_level_name
1.0.0.0.0	Data Centre
1.1.0.0.0	Site Preparation
1.1.1.0.0	Site Survey and Soil Testing
1.1.1.1.0	Geotechnical Analysis
1.1.1.1.1	Soil Sample Collection
1.1.1.1.2	Laboratory Testing
1.1.1.1.3	Geotechnical Report Preparation

Figure 43: Sample rows of a fictitious WBS table

PMBI uses two fields from your WBS table:

Column	Data Type	Description
WBS_ID	nvarchar	A unique identifier (5 integers delimited by dots) for each WBS node, structured to reflect its position in the hierarchy (e.g., 1.0.0.0.0, 1.1.0.0.0, 1.1.2.0.0…1.1.2.1.1…2.0.0.0.0, etc)
WBS_level_name	nvarchar	The display name for the WBS node as it will appear in reports and drill-down visuals

The WBS_ID coding scheme defines the parent-child relationships across the hierarchy. PMBI uses these relationships to roll up costs, earned value, and schedule performance from work packages at the bottom of the structure all the way to the project summary at the top. The drill-down navigation in the reporting visuals follows this hierarchy directly, so the accuracy of your WBS_ID values has a direct impact on how your data is organized and displayed.

Mapping Your Existing Structure to PMBI's WBS Format

If your organization already tracks project scope using a hierarchical WBS with dot-notation coding, the mapping process is straightforward. Each existing node translates directly to a WBS_ID, and the WBS_level_name field can carry your existing node names verbatim. In most cases this is simply a matter of verifying that your existing codes conform to the dot-delimited format before loading the data. The hierarchy itself does not need to change.

If your project hierarchy goes deeper than five levels, you will need to truncate your structure to five levels and roll up your lower-level project data to Level 5.

If your project has fewer than five levels, simply pad the unused lower levels with zeros to complete the ID. For example, a three-level structure would place the project at 1.0.0.0.0, a major scope area at 1.1.0.0.0, and a work package at 1.1.1.0.0. If you have unused levels after your work packages, you can also re-purpose them for tailored reporting rather than leaving them padded with zeros.

If your organization uses a different coding structure such as accounting cost codes, asset type classifications, procurement categories, or any other flat or semi-structured numbering system, some additional preparation will be needed, but the underlying logic remains the same. The key step is identifying the natural levels of grouping already present in your data and using those to construct the hierarchy. For example, if your cost codes already distinguish between major scope areas, sub-areas, and individual line items, those three tiers map directly to WBS levels 2, 3, and 4 respectively, with your project sitting at Level 1. Once you can see the hierarchy implicit in your existing structure, assigning dot-notation IDs that reflect it is typically a straightforward exercise in a spreadsheet.

If your organization does not currently use a WBS or other classification factor, it is worth taking the time to define a proper WBS before loading data into PMBI. A practical starting point is to ask: *what are the major physical or functional outcomes this project needs to deliver?* Those become your Level 2 nodes. Each one is then broken down into its component deliverables at Level 3, and so on until you reach a level of detail where work can be meaningfully planned, measured, and controlled. For construction projects, this often aligns naturally with physical areas, systems, or contract packages. Engaging your project controls lead or a scheduling professional to define this structure before data preparation begins will pay dividends in the quality and interpretability of your EVM reporting throughout the project.

Things to keep in mind:

When preparing your WBS table, keep the following in mind:

The WBS should reflect deliverable-oriented groupings, not process steps or organizational units. Each node should represent a tangible outcome or component of work, named using nouns and adjectives rather than action verbs.

PMBI supports up to five (5) levels of WBS hierarchy. Level 1, the top node, the project root or root level (#.0.0.0.0) represents the overall project and is represented by a number in the first position followed by zeros. Levels 2 , 3 and 4 (#.#.#.#.#) represent progressively finer breakdowns of scope. Level 5 (#.#.#.#.#) represents tasks or work packages. Typically, this lowest level of the WBS is where tasks are assigned, planned value and actuals are recorded, and EVM metrics are calculated. But PMBI does allow you to store project performance data at any level.

Not every branch of the WBS needs to be decomposed to the same depth. Some areas of the project may be well defined and broken down to Level 5, while others may remain at a higher level if the detail is not yet available or not required for control purposes.

Pitfalls to Avoid

Inconsistent ID coding. The WBS_ID field drives all hierarchy logic in PMBI. If IDs are formatted inconsistently (mixed delimiters, irregular zero-padding, or gaps in the numbering) nodes may not roll up correctly or may appear out of order in reports. Establish your ID format at the outset and apply it uniformly across the entire structure.

Mixing deliverables and activities. A common mistake is populating the WBS with task descriptions or process steps (e.g., "Review drawings", "Submit RFI") instead of deliverables (e.g., "Issued-for-Construction Drawings", "Design Approval"). Activity-based WBS structures undermine EVM calculations and make it difficult to measure percent complete in a meaningful way.

Excessive granularity at lower levels. Work packages that are too small in scope or too short in duration create reporting noise without adding management value. As a general rule, a work package should represent a meaningful and measurable unit of deliverable output.

Missing nodes in the hierarchy. Every child WBS node must have a valid parent in the structure. Orphaned nodes (where the parent ID does not exist in the table) will cause rollup errors and gaps in your reporting hierarchy. Always validate that the full chain from each work package back to the project root is present in the table before committing data.

Changing IDs after the project is underway. WBS_IDs are used as keys to link budget, actuals, planned value, and task data. Renaming or restructuring IDs mid-project will break those linkages between periods. If the WBS needs to be restructured after work has begun, this should be treated as a formal scope change and all related tables updated consistently.

Missing Project Root nodes in the hierarchy. Every project must have a root WBS node (i.e. 1.0.0.0.0).

A1.2 Budget Table

Budget_WBS_ID	Budget_original
[nvarchar]	*[decimal]*

Figure 44: The Budget table columns and data type

The Budget table defines the original approved budget for each WBS element. Each row links a budget amount to a specific node in your WBS hierarchy using the same WBS ID coding scheme established in your WBS table.

Column	Data Type	Description
Budget_WBS_ID	nvarchar	The WBS ID of the element this budget amount belongs to. Must match a WBS_ID value in your WBS table
Budget_original	decimal	The original approved budget for this WBS element, in your project's base currency

Budget is typically assigned at the work package level (the lowest level of your WBS) though it can be assigned at higher levels if detailed task-level planning are not warranted. The sum of all budget rows should equal the total authorized project budget, which will appear as the **Current Budget** figure in your project summary before any budget adjustments are applied.

Things to keep in mind:

Each WBS ID in this table must exist in your WBS table. Budget rows referencing IDs that do not appear in the WBS will be rejected during the commit process.

Budget_WBS_ID	Budget_original
1.1.1.1.1	$725,000.00
1.1.1.1.2	$37,000.00
1.1.1.1.3	$3,500.00

Figure 45: Sample rows of a fictitious Budget table

A1.3 Tasks Table

Task_name	Task_ID	Task_EAC	Task_WBS_ID
[nvarchar]	*[nvarchar]*	*[decimal]*	*[nvarchar]*

Figure 46: The Tasks table columns and data type

The Tasks table defines the individual work packages that make up your project's scope. Each row represents a single task and links it to its position in the WBS hierarchy. Tasks are the grain at which planned value, actuals, and earned value are recorded. All EVM metrics ultimately trace back to the task level.

Column	Data Type	Description
Task_name	nvarchar	The display name of the task as it will appear in the Task Details table and other report visuals
Task_ID	nvarchar	A unique identifier for this task. Used to link records in the Actuals and Planned Value tables back to this task
Task_EAC	decimal	The current Estimate at Completion for this task
Task_WBS_ID	nvarchar	The WBS ID of the work package this task belongs to. Must match a WBS_ID value in your WBS table

The Task_ID value is the key that connects this table to your Actuals and Planned Value data. Every actual cost entry and every planned value entry must reference a valid Task_ID from this table. Tasks that exist here but have no corresponding Actuals or Planned Value rows will appear in the Task Details table with zero values across all EVM metrics and will show as 0% complete.

The Task_WBS_ID must match a WBS_ID in your WBS table. This is how PMBI knows where in the hierarchy each task sits and how to roll its performance up through the WBS levels to the project summary.

Things to Keep in Mind

Task_EAC represents your project's current forecast, not the original budget. As the project progresses and conditions change, this value should reflect the most current estimate of what the task will ultimately cost to complete. The relationship between Task_EAC and the original budget assigned in the Budget table is what drives the cost variance and overrun signals visible in your reporting.

Task_ID values must be unique across the table. Duplicate Task_IDs will cause a warning to appear during the commit process.

As with the WBS and Budget tables, Task_WBS_ID values must reference nodes that exist in your WBS table. Tasks pointing to WBS IDs that do not exist will be rejected during commit.

Task_name	Task_ID	Task_EAC	Task_WBS_ID
Bore Holes - sector 1	T-0001	$177,686.54	1.1.1.1.1
Bore Holes - sector 2	T-0002	$251,196.42	1.1.1.1.1
Bore Holes - sector 3	T-0003	$145,221.88	1.1.1.1.1
Bore Holes - sector 4	T-0004	$205,284.35	1.1.1.1.1
Lab Testing - sector 1	T-0005	$8,915.64	1.1.1.1.2
Lab Testing - sector 2	T-0006	$12,000.07	1.1.1.1.2
Lab Testing - sector 3	T-0007	$7,738.07	1.1.1.1.2
Lab Testing - sector 4	T-0008	$9,903.58	1.1.1.1.2
Final Report	T-0009	$3,558.84	1.1.1.1.3

Figure 47: Sample rows of a fictitious Tasks table

A1.4 Planned Value (PV) Table

Task_name	Task_ID	Task_EAC	Task_WBS_ID
[nvarchar]	*[nvarchar]*	*[decimal]*	*[nvarchar]*

Figure 48: The PV table columns and data type

The Planned Value (PV) table defines the time-phased budget baseline for each task. In other words, how much work was planned to be accomplished in each reporting period. Each row represents a single period's planned value for a specific task, and together these rows form the S-curve baseline against which actual progress is measured.

Column	Data Type	Description
PV_task_ID	nvarchar	The Task ID this planned value entry belongs to. Must match a Task_ID value in your Tasks table
PV_period	date	The period end date this planned value entry applies to
PV_amount	decimal	The budgeted value of work planned to be completed in this period for this task

Relationship to Other Tables

Each PV_task_ID must reference a valid Task_ID in your Tasks table. Planned value entries for tasks that do not exist in the Tasks table will be rejected during the commit process.

The cumulative sum of all PV_amount values for a given task represents the total planned value for that task over its full duration.

The Relationship Between PV and Budget

The total cumulative planned value across all tasks should equal the total budget assigned in your Budget table. When these two figures align, the Budget Allocation Ratio will read 1.00, meaning the full authorized budget is accounted for in the task plan.

If the cumulative total of your PV table is less than your total budget, the Budget Allocation Ratio will fall below 1.00. This indicates that a portion of the authorized budget has not yet been allocated to specific tasks, which may reflect scope not yet planned, budget intentionally held in reserve, or budget approved but not yet broken down into the task structure.

Conversely, if cumulative PV exceeds the total budget, the Budget Allocation Ratio will rise above 1.00, indicating that tasks have been planned for more than the currently authorized budget. This warrants investigation to confirm whether a budget adjustment is required.

Things to Keep in Mind

PV should be entered at the period level, not loaded as a single lump-sum row per task.

Every task that has a budget assigned should have at least one PV row. Tasks with a budget but no planned value entries will not contribute to the S-curve baseline and will appear to have no scheduled work, which will affect schedule performance calculations at both the task and project level.

PV_task_ID	PV_period	PV_amount
T-0172	2024-11-30	$25,000.00
T-0164	2024-12-31	$15,000.00
T-0172	2024-12-31	$25,000.00
T-0164	2025-01-31	$20,000.00
T-0165	2025-01-31	$15,000.00
T-0172	2025-01-31	$25,000.00
T-0164	2025-02-28	$13,500.00

Figure 49: Sample rows of a fictitious PV table

A1.5 Actuals Table

Actual_expense	Actual_reporting_date	Actual_EV	Actual_task_ID
[decimal]	[date]	[decimal]	[nvarchar]

Figure 50: The Actuals table columns and data type

The Actuals table is where real project activity is recorded including the costs incurred and the value earned, period by period, for each task. Together with the Planned Value table, it is the primary source of all EVM performance metrics in PMBI.

Column	Data Type	Description
Actual_expense	decimal	The actual cost incurred for this task in this reporting period
Actual_reporting_date	date	The period end date this entry applies to
Actual_EV	decimal	The earned value recognized for this task in this reporting period (the budgeted value of work actually completed)
Actual_task_ID	nvarchar	The Task ID this entry belongs to. Must match a Task_ID value in your Tasks table

The distinction between Actual_expense and Actual_EV is fundamental to EVM and is worth understanding clearly before loading data.

Actual_expense is money spent or commonly referred to Actual Cost. It answers the question: *what did we pay for work on this task this period?*

Actual_EV is a measure of how much work was actually completed this period, expressed in dollar terms using the original budget as the measuring stick. It answers the question: *based on the portion of this task we finished this period, what fraction of the task's budget did we earn?*

To make this concrete: if a task has a total budget of $100,000 and your team completed 20% of it this period, your Actual_EV for that period is $20,000 — regardless of whether you actually spent more or less than that to get there. The $20,000 is not money paid out; it is the dollar value of the progress made, measured against what was originally planned for that portion of work.

This is why EV is such a powerful project management signal. It gives you a common unit to compare what you planned to accomplish, what you actually spent, and what you actually got done. Without it, you would only know how much you spent, with no way to objectively measure whether that spending produced the expected amount of completed work.

How Actual_EV is calculated in practice depends on the earning technique applied to each task. Common approaches include fixed milestones, percent complete, and units complete, among others. These are discussed further later in the Appendix (A2).

The relationship between these two columns is what produces your cost performance signal. When Actual_expense exceeds Actual_EV for a given period, the project is spending more than the value it is producing. This is a cost overrun condition reflected in a CPI below 1.00. When Actual_EV exceeds Actual_expense, the project is producing more value than it is spending. This is a favorable cost position reflected in a CPI greater than 1.00.

Multiple Rows Per Task Per Period

Unlike the Planned Value table, the Actuals table legitimately allows more than one row for the same task in the same reporting period. This reflects real-world practice where multiple cost entries from different subcontractors, cost codes, or invoices may be recorded against a single task within the same period.

The sample data below illustrates this. Task T-0172 has two entries in different periods, and task T-0164 has two entries in the same period (January 2025), both of which are valid and will be aggregated:

Actual_expense	Actual_reporting_date	Actual_EV	Actual_task_ID
$15,038.28	2024-10-31	$14,640.95	T-0172
$24,779.53	2024-11-30	$23,657.37	T-0172
$14,790.85	2025-01-31	$14,144.14	T-0164
$25,724.87	2024-12-31	$23,858.04	T-0172
$18,734.75	2025-01-31	$17,612.22	T-0164
$14,298.86	2025-01-31	$13,460.76	T-0165

Figure 51; Sample rows of a fictitious Actuals table

Things to Keep in Mind

Actual_task_ID must reference a valid Task_ID in your Tasks table. Entries pointing to tasks that do not exist will be rejected during the commit process.

Actual_EV should never be loaded as a copy of Actual_expense. These are distinct measurements and conflating them will eliminate all cost performance signal from your reporting. If you are unsure how to calculate earned value for your tasks, refer to Appendix on Earned Value (A2) for guidance on common EV techniques and when to apply them.

Period dates should be consistently formated across the Actuals and Planned Value tables.

A1.6 Budget Adjustments Table

Budget_adj_change_ID	Budget_adj_amount	Budget_adj_WBS_ID
[nvarchar]	*[decimal]*	*[nvarchar]*

Figure 52: The Budget Adjustments table columns and data type

The Budget Adjustments table displays approved changes to the original project budget. Each row represents a single budget change event and links it to the WBS element it affects. Together with the Budget table, it forms the complete picture of the project's current authorized budget.

Column	Data Type	Description
Budget_adj_change_ID	nvarchar	A reference identifier for this budget change. This is not an application-specific value. You may use whatever change ID or reference number your organization already uses internally, such as a change order number, PCO reference, or project control log ID
Budget_adj_amount	decimal	The dollar amount of this budget change. Use a positive value for budget increases and a negative value for budget reductions
Budget_adj_WBS_ID	nvarchar	The WBS element this adjustment applies to. Must match a WBS_ID value in your WBS table

PMBI combines the original budget from the Budget table with all applicable adjustments from this table to arrive at the current authorized budget for each WBS element.

A single WBS element can have multiple adjustment rows. For example, if a scope change was approved in phases or if multiple change orders apply to the same area of work. PMBI will sum all

adjustment rows for a given WBS element to calculate the net adjustment displayed in the Budget Changes page.

Things to Keep in Mind

Budget_adj_change_ID is a free-form reference field. It can represent a signed change order, an approved PCO, a project controls log entry, or any other internal reference your organization uses. PMBI does not validate or enforce any particular format, so use whatever naming convention is most meaningful for your project controls process.

Budget_adj_WBS_ID must reference a WBS_ID that exists in your WBS table. Adjustments pointing to WBS elements that do not exist will be rejected during the commit process.

If your project has no approved budget changes, this table can be left empty.

Budget_adj_change_ID	Budget_adj_amount	Budget_adj_WBS_ID
CO_1	$1,000,000.00	5.6.3.0.0

Figure 53: Sample rows of a fictitious Budget Adjustment table

A2 – Earned Value

What Is Earned Value?

Earned Value (EV) is the budgeted value of work actually completed at a point in time. It is not what you spent, and it is not what you planned to spend. It is the portion of your project budget that you have legitimately "earned" through completed work. This distinction is what makes EV a powerful project controls tool: it gives you a single, objective measure of progress expressed in dollars, allowing you to compare what was planned, what was spent, and what was actually accomplished using a common unit.

The challenge EV presents to most project teams is not understanding the concept but rather determining how to calculate it consistently and defensibly across all tasks in a project. The method used to translate physical progress into an earned value dollar amount is called an earning technique, and choosing the right technique for each type of work is one of the most important decisions in setting up a sound EVM framework.

Common Earning Techniques

Several earning techniques are widely used in practice, each suited to different types of work:

Independent Verification uses a third party or an independent internal reviewer to assess and confirm the percentage of work completed on a task before earned value is recognized. Rather than relying on the team performing the work to self-report progress, a site inspector, quality assurance reviewer, or project controls professional conducts a formal assessment and certifies the completion status. This technique adds a layer of objectivity and accountability that is particularly valuable on

high-value work packages, contractually sensitive scope, or any situation where self-reported progress estimates carry meaningful financial or schedule implications. It is common in owner-managed capital projects, infrastructure programs, and contracts where milestone payments are tied directly to verified progress.

Fixed Formula (e.g., 0/100, 25/75, 50/50) assigns earned value at predefined milestones rather than continuously. A 0/100 rule earns nothing until the work package is complete, at which point 100% of the budget is earned. A 50/50 rule earns half the budget when work begins and the other half upon completion. These techniques are simple to apply and eliminate subjectivity, making them well suited to short-duration tasks where interim progress measurement would add little value.

Percent Complete earns value proportionally based on an assessed percentage of completion. If a task has a budget of $100,000 and is assessed as 40% complete, the earned value is $40,000. This technique is flexible and intuitive but introduces subjectivity, since the percent complete estimate relies on someone's judgment. It works best when there are observable, verifiable indicators of progress that reduce reliance on opinion.

Units Complete measures progress by counting discrete, identical units of output against a total quantity. If a task involves installing 500 light fixtures and 200 have been installed, the task is 40% complete and EV is calculated accordingly. This is one of the most objective techniques available and is highly reliable wherever work can be broken into countable, uniform units.

Weighted Milestones divides a work package into a series of defined milestones, each assigned a portion of the total budget based on its relative significance. Earned value is recognized when each milestone is achieved. This technique works well for longer-duration tasks that pass through identifiable stages, providing periodic objective measurement without requiring continuous progress assessment.

Level of Effort (LOE) is a special case used for support activities whose value is tied to the passage of time rather than the production of a discrete deliverable. EV for LOE tasks is earned simply by virtue of time elapsed, proportional to the planned schedule. It is appropriate only for tasks such as ongoing project administration, management oversight, or support services where no measurable output exists. Because LOE earned value is driven entirely by the planned schedule, a re-baseline event requires careful attention. If the project schedule is extended or compressed and the LOE task's planned value profile is updated to reflect the new timeline, the earned value calculation will shift accordingly, which can produce sudden changes in reported performance that reflect the re-baseline rather than any real change in work accomplished. When re-baselining, it is good practice to document the change clearly and communicate to stakeholders that any movement in LOE performance metrics at that point in time is a consequence of the schedule adjustment rather than a genuine improvement or deterioration in project conditions.

Where EV Is Easier and Where It Gets Harder

Some work lends itself naturally to objective EV measurement. Construction and installation activities with defined quantities, clear deliverables, and visible physical progress are the easiest cases. A concrete pour, a structural steel erection package, or a piping installation with a defined linear footage all have countable, verifiable outputs that make earning straightforward and defensible.

The difficulty increases with work that is more cognitive, iterative, or administrative in nature. Engineering and design tasks, for example, can be partially complete in ways that are genuinely hard to assess. A drawing set that is 80% drafted may still require 50% of the remaining effort if the most complex details are unresolved. Project administration, management time, and people costs present a similar challenge because the "output" is diffuse and continuous rather than discrete and countable. For these categories, organizations often default to LOE or a fixed formula, accepting that precision is less achievable and that the goal is a reasonable, consistent approximation rather than a precise measurement.

The key principle is that EV should never be based on costs incurred. Spending money is not the same as completing work, and using expenditure as a proxy for progress is one of the most common errors in EVM practice. It eliminates all cost performance signal and guarantees a CPI of 1.00 regardless of true project conditions.

Examples Across Industries

Construction: A road resurfacing project includes a work package for asphalt paving with a budget of $500,000 covering 10 kilometers of roadway. At the reporting date, 4 kilometers have been completed. Using units complete, the earned value is $200,000, representing 40% of the budget for 40% of the physical work done.

Engineering and Design: A process engineering firm is producing a set of 20 piping and instrumentation diagrams (P&IDs) under a work package budgeted at $80,000. Each diagram is assigned equal weight. At the reporting date, 12 diagrams have been issued for review. Using weighted milestones with each diagram representing one twentieth of the budget, the earned value is $48,000.

Information Technology: A software implementation project includes a user acceptance testing work package budgeted at $60,000, structured around three defined milestone gates: test environment ready (20%), test execution complete (60%), and sign-off received (100%). At the reporting date, the first two milestones have been achieved, earning $48,000 of the $60,000 budget.

Project Administration: An owner's project management team is providing oversight services across a 24-month project under a work package budgeted at $240,000, planned evenly at $10,000 per month. Using level of effort, earned value is recognized at $10,000 per month as each month elapses, regardless of the volume of meetings, correspondence, or decisions occurring in any given period. After 10 months, earned value stands at $100,000.