

Power Bi – P6 App

USER MANUAL

Khono Project Control Services

No.22, Corner 2nd St & Voortrekker Ave, Edenvale, Johannesburg

Contents

1. Overview	2
2. Installation and Setup.....	2
3. Loading and Transferring Data	3
1- Select Database.....	3
2- Select Project(s)	3
3- Run	4
4. Setting up Relationships	5
5. Useful Formulas	6

1. Overview

Developing reports and dashboards from project schedules is a common practice among project controls professionals, particularly planners. However, preparing data, copying and pasting it into Excel, and connecting it to Power BI can be a tedious process. This becomes even more time-consuming when dashboards require weekly or monthly updates.

The **KhonoPowerBiP6 App** is designed to help project controls professionals develop and update dashboards more quickly and with less effort, saving valuable time that can be spent on other important project controls tasks.

The app reads data directly from the Primavera P6 SQLite database, eliminating the need for Excel as an intermediary and reducing the need for data transformation in Power BI.

Once a dashboard has been developed, updating it is straightforward: use the app to read the updated schedule and replace the existing exported data, then refresh the data in Power BI.

Important: Primavera P6 uses complex calculations to determine remaining durations, remaining units, and remaining costs, particularly at the WBS level. The app does not currently replicate every aspect of these calculations (that is almost impossible!). Consequently, for complex schedules containing many out-of-sequence activities, constraints, and Level of Effort activities, some results may differ from those produced by P6. We have tested multiple schedules and simulated P6's calculation processes across various scenarios; however, not every possible scenario has been covered.

When the app is first run on a computer, an email address is required to activate the **14-day free trial**. After the trial expires, you must purchase a licence to continue using the app. Each licence is valid for one computer and cannot be used on another computer.

For any support, contact: info@khonopc.com

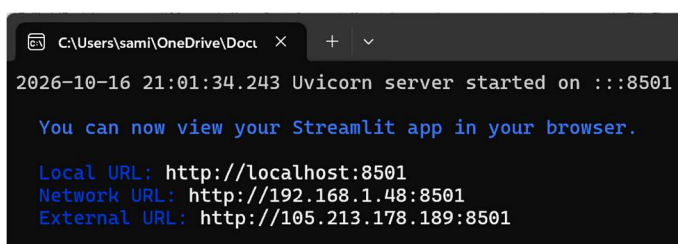
2. Installation and Setup

After downloading the application's ZIP file, extract its contents to a folder on your PC. If necessary, you can copy or move the extracted folder to another location. The extracted folder contains the following items:

- _internal
- P6PowerBI_App.exe
- KhonoP6PowerBi App User Manual

To install the application, right click on the P6PowerBI_App.exe file, then select Open or run it by double clicking. The App will run and opens two following Window and tab.

- A tab in a browser (Edge or Chrome), that will show the application.
- Command Window that is similar to the following screenshot. It must remain open while the application is running. You can close it when you have finished using the application



```
C:\Users\sami\OneDrive\Docu X + v
2026-10-16 21:01:34.243 Uvicorn server started on ::8501

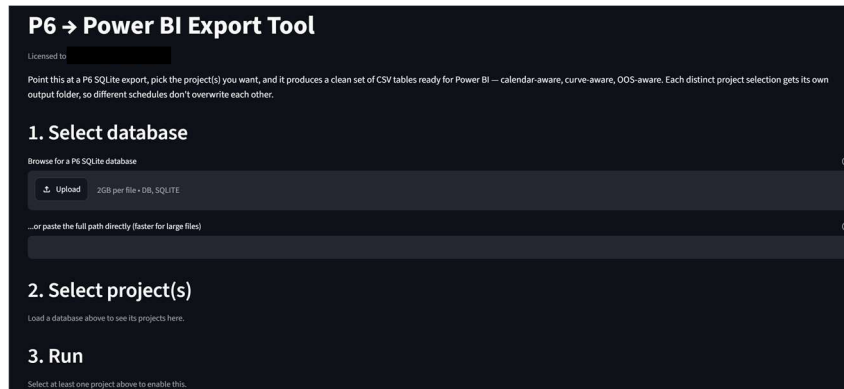
You can now view your Streamlit app in your browser.

Local URL: http://localhost:8501
Network URL: http://192.168.1.48:8501
External URL: http://105.213.178.189:8501
```

Note: Always start the application by running P6PowerBI_App.exe.

3. Loading and Transferring Data

After registering the application, either through the free trial or a purchased license, the application page will display three steps, as shown in the following screenshot:



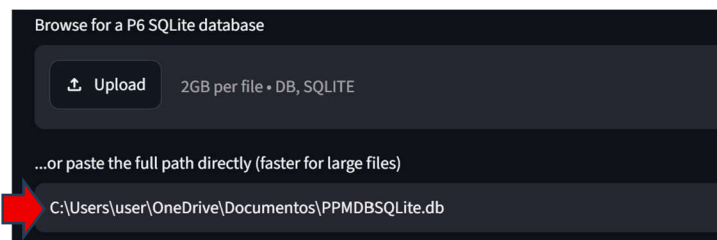
1- Select Database

In this step, you select the Primavera P6 database from which the schedules will be loaded. Click the **Upload** button, browse to the location of the P6 database, and select it.

If the database is larger than 2 GB, the upload process will not work. In this case, you must enter the database path manually in the box below the Upload button. To obtain the database path:

- Locate the database file in File Explorer.
- Right-click the file and select Copy as path.
- Paste the path into the box below the Upload button.
- Remove the quotation marks (") from the beginning and end of the path.

The path should then appear as illustrated below.



2- Select Project(s)

After selecting the database, click **Load projects from this database**. The app will load the projects available in the selected database.

Click **Choose options** and select the schedule or schedules required for dashboard development. You can select multiple schedules as needed.

Note: If required, select both the baseline and the updated schedule so that the data from both schedules can be used in the dashboard.

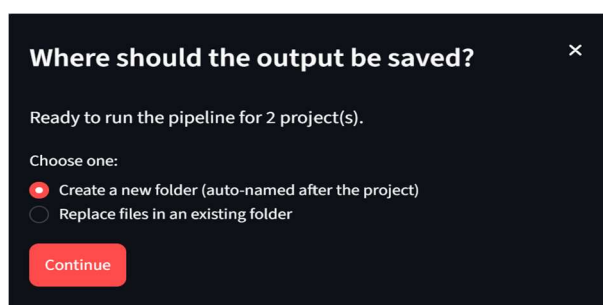
3- Run

After selecting the required schedule(s), click **Run**.

The **Where should the output be saved?** window will appear.

Two options are available:

- **First option (default):** Select this option when exporting a schedule for the first time, or when creating a new dashboard and no previous export data needs to be replaced.
- **Second option — Replace files in an existing folder:** Select this option when updating an existing dashboard with data from an updated schedule. The app will replace the previously exported schedule data with the newly generated data.



When the second option is selected, a drop-down list will appear. From the list, select the folder that its files should be replaced.

Note: Before replacing existing schedule data, make a backup copy of the relevant folder. Rename the backup to include identifying information such as the data date, project name, and backup date. This will allow you to restore the previous data if necessary.

Click **Continue** to start the export process.

The app will generate CSV files and save them in a folder named after the selected schedule. Once the process has completed successfully, a confirmation message will be displayed.

The schedule folder will be created inside the **output** folder, located in the same directory as P6PowerBI_App.exe and the _internal folder.

Each project folder will contain **15 CSV files**.

Tables produced		
Dim_Project	Dim_Activity	Dim_WBS
1 rows	2,630 rows	1,481 rows
Bridge_WBS_Rollup	Dim_ActivityCode	Bridge_Task_ActivityCode
11,054 rows	854 rows	33,167 rows
Dim_Resource	Fact_Assignment	Calendars_in_use
89 rows	2,104 rows	7 rows
Dim_Curve	Fact_Assignment_Daily	Dim_Calendar_Day
1 rows	27,197 rows	6,552 rows
Fact_Actual_Daily	Fact_EarnedValue_Daily	Fact_Remaining_Daily
0 rows	0 rows	27,240 rows

Important: The data in the csv files are ready to use and no need to make any process, data transform in Power Bi.

4. Setting up Relationships

In Power BI, create a new report and import all the CSV files from the relevant project folder.

During data import, Power BI may automatically create relationships between the tables. Delete the automatically generated relationships and create the required relationships according to the instructions provided below.

- Dim_project to Dim_WBS: Link project_id to project_id, one-to-many relationship
- Dim_WBS to Bridge_WBS_Rollup: Link wbs_id to ancestor_wbs_id, one-to-many relationship
- Bridge_WBS_Rollup to Dim_Activity: Link descendant_wbs_id to wbs_id, one-to-many relationship. If not possible, set it to many-to-many.
- Dim_Activity to Fact_Actual_Daily: Link task_id to task_id, one-to-many relationship
- Dim_Activity to Fact_Assignment: Link task_id to task_id, one-to-many relationship
- Dim_Activity to Fact_Assignment_Daily: Link task_id to task_id, one-to-many relationship
- Dim_Activity to Fact_EarnedValue_Daily: Link task_id to task_id, one-to-many relationship
- Dim_Activity to Fact_Remaining_Daily: Link task_id to task_id, one-to-many relationship
- Dim_Resource to Fact_Actual_Daily: Link task_id to task_id, one-to-many relationship
- Dim_Resource to Fact_Assignment: Link task_id to task_id, one-to-many relationship
- Dim_Resource to Fact_Assignment_Daily: Link task_id to task_id, one-to-many relationship
- Dim_Resource to Fact_EarnedValue_Daily: Link task_id to task_id, one-to-many relationship
- Dim_Resource to Fact_Remaining_Daily: Link task_id to task_id, one-to-many relationship
- Dim_Date to Fact_Assignment: Link task_id to task_id, one-to-many relationship
- Dim_Date to Fact_Assignment_Daily: Link task_id to task_id, one-to-many relationship
- Dim_Date to Fact_EarnedValue_Daily: Link task_id to task_id, one-to-many relationship
- Dim_Date to Fact_Remaining_Daily: Link task_id to task_id, one-to-many relationship

Note: Create a new table in Power Bi and name it Dim_Date and apply the following formula:

```
Dim_Date =
CALENDAR(
    MIN( Fact_Assignment_Daily[date] ),
    MAX( Fact_Assignment_Daily[date] )
)
```

- Dim_ActivityCode to Bridge_Task_ActivityCode: Link actv_code_id to actv_code_id, one-to-many relationship
- Bridge_Task_ActivityCode to Dim_Activity: Link task_id to task_id, one-to-many relationship

- Dim_Project to Fact_Assignment: Line task_id to task_id, one-to-many relationship
- Dim_Project to Fact_Assignment_Daily: Line task_id to task_id, one-to-many relationship
- Dim_Project e to Fact_EarnedValue_Daily: Line task_id to task_id, one-to-many relationship
- Dim_Project to Fact_Remaining_Daily: Line task_id to task_id, one-to-many relationship

Note: When more than one schedule being loaded, learn the proj_id of loaded projects by opening the Dim_project in the Table view. Proj_id is critical for writing formulas.

5. Useful Formulas

Use the following formulas for your calculations.

To use a formula, create a **New Measure** in Power BI and assign it the suggested name.

In each formula, replace the proj_id value with the appropriate project ID from the Dim_Project table.

- **Actual Total Cost**

```
Actual Total Cost =
CALCULATE(
    SUM(Fact_Actual_Daily[daily_act_cost]),
    Fact_Actual_Daily[proj_id] = 1094
)
```

- **Baseline Finish Date**

```
Baseline Finish Date =
CALCULATE(
    MAX(Dim_Activity[target_end_date]),
    Dim_Activity[proj_id] = 1147
)
```

- **Baseline Start Date**

```
Baseline Start Date =
CALCULATE(
    MIN(Dim_Activity[target_start_date]),
    Dim_Activity[proj_id] = 1147
)
```

- **Current Finish Date**

```
Current Finish Date =
CALCULATE(
    Max(Dim_Activity[early_end_date]),
    Dim_Activity[proj_id] = 1094
)
```

- **Data Date**

```
Data Date =
CALCULATE(
    Max(Dim_Project[data_date]),
    Dim_Project[proj_id] = 1094
)
```

- **Earned Value to Date**

```

Earned Value to Date =
VAR CutoffDate = [Data Date]
RETURN
    CALCULATE(
        SUM(Fact_EarnedValue_Daily[daily_ev_cost]),
        Fact_EarnedValue_Daily[proj_id] = 1094,
        Fact_EarnedValue_Daily[date] <= CutoffDate
    )

```

- **Planned Cost to Date**

```

Planned Cost to Date =
VAR CutoffDate = [Data Date]
RETURN
    CALCULATE(
        SUM(Fact_Assignment_Daily[daily_cost]),
        Fact_Assignment_Daily[proj_id] = 1094,
        Fact_Assignment_Daily[date] <= CutoffDate
    )

```

- **Cumulative Earned**

```

Cumulative Earned =
VAR CutoffDate = [Data Date]
VAR CurrentAxisDate = MAX ( Dim_Date[Date] )
RETURN
    IF (
        CurrentAxisDate <= CutoffDate,
        CALCULATE(
            SUM ( Fact_EarnedValue_Daily[daily_ev_cost] ),
            Fact_EarnedValue_Daily[proj_id] = 1094,
            FILTER(
                ALLSELECTED ( Dim_Date[Date] ),
                Dim_Date[Date] <= CurrentAxisDate
            )
        ),
        BLANK ()
    )

```

- **Cumulative Planned**

```

Cumulative Planned =
CALCULATE(
    SUM(Fact_Assignment_Daily[daily_cost]),
    Fact_Assignment_Daily[proj_id] = 1147,
    FILTER(
        ALLSELECTED(Dim_Date[Date]),
        Dim_Date[Date] <= MAX(Dim_Date[Date])
    )
)

```