\*\*User Guide: How to Process and Upload Your Dashboard Data\*\*

---

\*\*Dashboard Assumptions:\*\*

1. An aggregate RSE inputs file has been processed to produce an "aggregate summary" output.

2. This output, with a CBR calculation, is named "CBR and Baseline Risk Reduction" and is stored in the "CBR output" folder.

3. If you didn't process the RSE aggregate file yourself, consult with the team to ensure you have the most recent version.

---

\*\*Preparing Your Data for Consumption:\*\*

1. \*\*Tables in the Aggregate Summary File:\*\* Confirm the presence of the following tables:

- \*\*Risk-Level Baseline RR:\*\* Extracts the risk name.

- \*\*TableRSEbatch:\*\* Extracts the PVRR value.

- \*\*Tranche-Level CBR, ProgramRisk-Level CBR & Program-Level CBR:\*\* For their respective CBR values.

2. \*\*Pivoting and Merging Data:\*\* This allows for comparisons, patterns, and yearly trend visualization.

- Use your preferred tool. In this guide, we'll utilize Python's Pandas library.

- Note: Pandas doesn't directly melt multiple columns with varying values. To do this, melt each Data Frame separately and concatenate the results.

3. \*\*Save your Data:\*\* For optimal compatibility with Foundry, save your data as a CSV file.

---

\*\*Uploading Data to Foundry:\*\*

1. Review the CBR file on your computer to ensure it's formatted as desired.

2. In Foundry, go to: `workflow/data/eorm/cbr/raw`.

3. To upload:

- Drag and drop the file into the "eorm cbr" folder.

- Alternatively, click the green '+new' button, then select "import file as a dataset".

After uploading:

- For Excel files, you'll see an "import Excel workbook" window. Select "Import as dataset" (recommended) and proceed.

- If prompted to apply a schema to convert the file into a dataset, do so.

After uploading the CBR file to Foundry, update the CBR pipeline, run the build, and submit a new Pull Request for merging to master.

---

\*\*Updating the CBR Pipeline in Foundry:\*\*

1. Navigate to `risk\_repos\_pipeline` and create a new branch from master.

2. Head to `transforms-python -> src -> eorm\_ra -> datasets -> cbr`. Click on the `cbr\_and\_baseline\_risk\_reduction` Python file to update the input dataset.

3. To find the dataset's Foundry ID, open the cbr dataset, go to the "Details" tab, and copy the "Dataset RID" from the "About" page.

4. Replace the pipeline's input data source with the Dataset RID you copied.

5. Click "commit" (upper right). A window will pop up for comment addition (for tracking). After commenting, click "build".

6. Once the build is done, click "Propose changes" to submit a new PR to merge the dataset to master.

7. Finally, rebuild on the master branch to access the dataset.

---

\*\*Connecting to Power BI:\*\*

1. Open Power BI.

2. Go to: `File -> Get Data`. Under "Get Data", select "get data".

3. In the upcoming window, type "Palantir" in the search bar and select "Palantir Foundry".

4. Input the base URL and the optional Dataset RID. Choose between "import" or "directQuery", then click "OK".

5. After importing the data to Power BI, you can begin crafting your reports.

P.S. Something to note about Import and DirectQuery in Power BI

Import: A copy of the data from the selected tables and columns imports into Power BI Desktop. As you create or interact with visualizations, Power BI Desktop uses the imported data. To see underlying data changes after the initial import or the most recent refresh, you must import the full dataset again to refresh the data.

DirectQuery: No data imports into Power BI Desktop. For relational sources, you can select tables and columns to appear in the Power BI Desktop Fields list. As you create or interact with visualizations, Power BI Desktop queries the underlying data source, so you're always viewing current data. However, there are some considerations and limitations with DirectQuery:

1. Some Power BI Desktop features aren't supported in DirectQuery mode, or they have limitations. Some capabilities in the Power BI service, such as quick insights, also aren't available for datasets that use DirectQuery.
2. DirectQuery defines a one-million row limit for data returned from cloud data sources, which are any data sources that aren't on-premises.

Quantified Risk Scores reports

Datasets needed:

Driver-Outcome-Level Risk Score

Driver-Subdriver-Level Risk Score

FN Curve

Risk-Level Risk Score

Tranche-Driver-Subdriver Risk Score

Tranche-Level Risk Score

Tranche-Outcome-Level Risk Score