# On Evolvability in Tools for Data Analytics and Intelligence: the Power BI case

Prof. Dr. ir Geert Haerens
Prof. Dr. ir. Herwig Mannaert

PATTERNS – 2026

Lisbao

# The Authors

Prof. Dr. ir Geert Haerens

Assistant Prof. @ Antwerp Mngt School
Post Doc @ Antwerp University

Enterprise Architect @ Engie

Prof. Dr. ir Herwig Mannaert

Antwerp University
Faculty of Applied Economics
Department Business Inform

Co-founder of NS theory & N

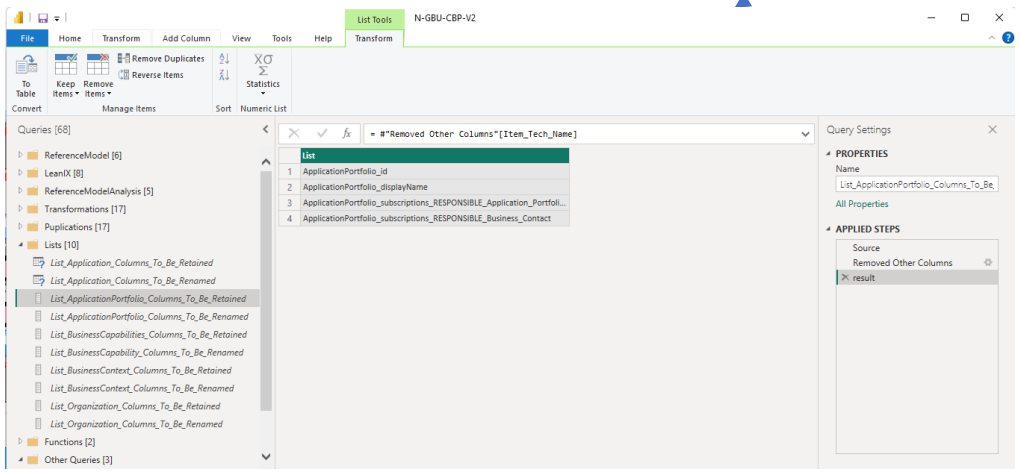
# Presentation Content

- **Research Introduction**
- **Introducing Power BI**
- **Measure Evolvability with Normalized Systems Theory**
- **Demonstrating non-evolvability of Power**
- **Discussion**
- **Conclusion**

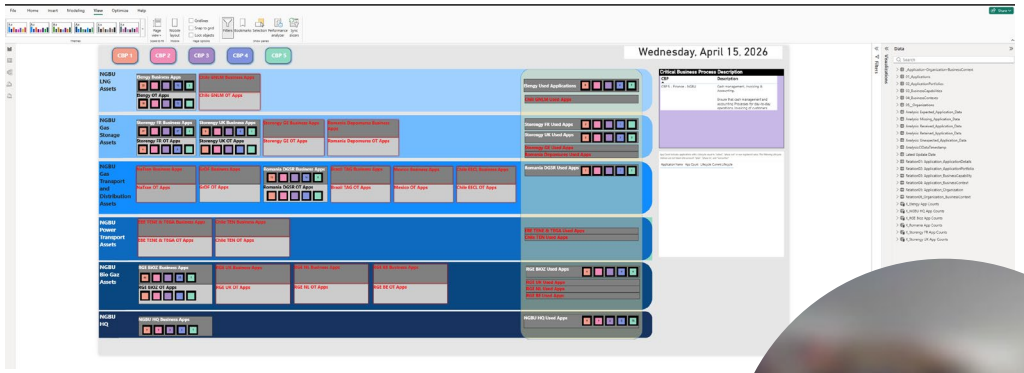
# Research Introduction


# Introducing Power BI

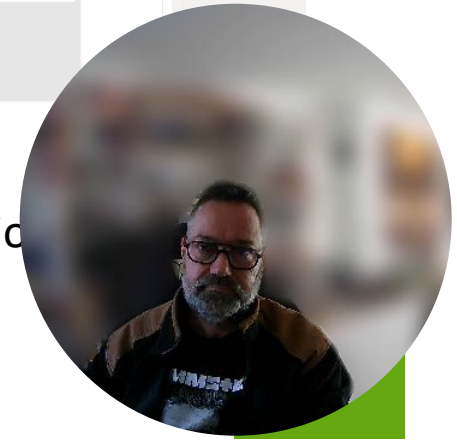
**Power BI**




Power Query: ETL via Low-Code/No Code

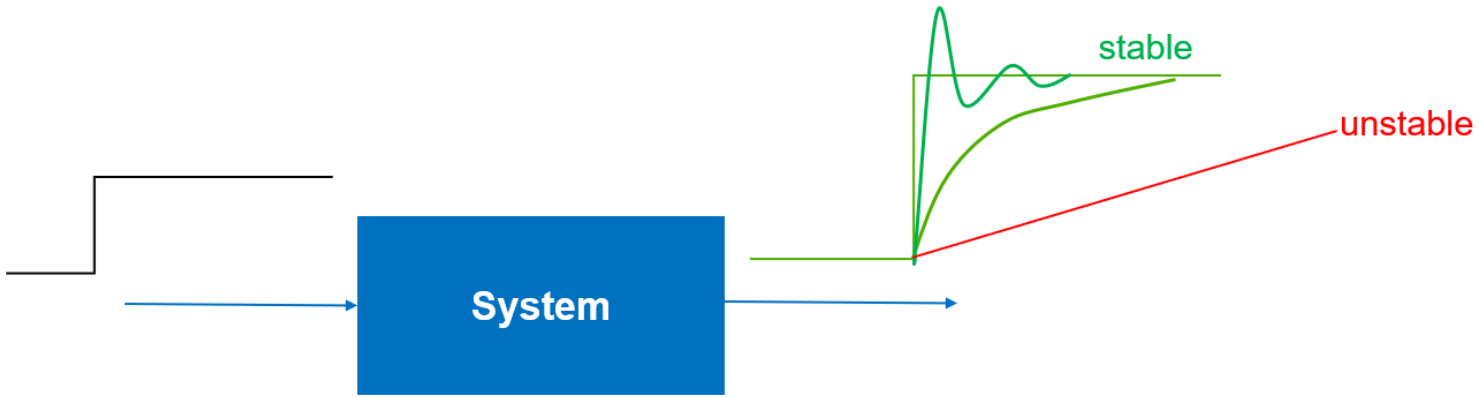



Power BI Desktop: Data Visualizatio

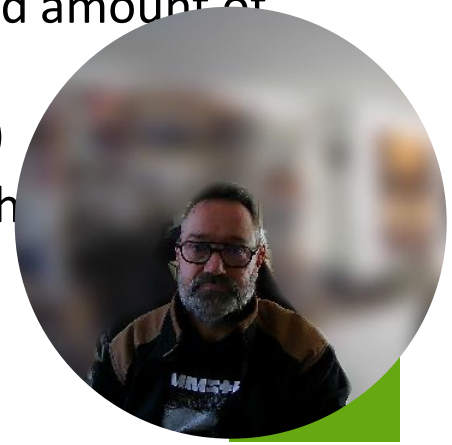


# Measure Evolvability with Normalized Systems Theory

- **From Classic Engineering :**
  - A system is considered stable when a bounded input results in a bounded output, aka BIBO




- **For System Design Changes:**
  - A system is considered stable when a bounded functional change results in a bounded amount of work and is independent of the size of the System.
  - When the size of the System has an impact, we talk about a Combinatorial Effect (CE)
  - A system must be free of Combinatorial Effects for it to be stable under change and th
  - EVOLVABILITY is measured by the ABSENCE of CE



# Measure Evolvability with Normalized Systems Theory

- **The following NS Theorems are the necessary conditions to be free of CE**

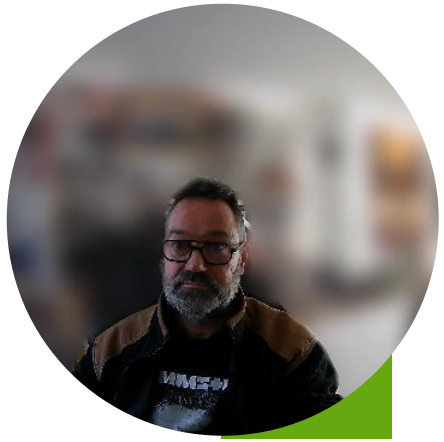
  - **SoC:** Separation of Concerns
    - *A processing function can only contain a single task in order to achieve stability.*
  - **DvT:** Data Version Transparency
    - A data structure that is passed through the interface of a processing function needs to exhibit version transparency in order to achieve stability.
  - **AvT:** Action Version Transparency
    - A processing function that is called by another processing function needs to exhibit version transparency in order to achieve stability.
  - **SoS:** Separation of State
    - *Calling a processing function within another processing function, needs to exhibit st keeping in order to achieve stability.*

# Measure Evolvability with Normalized Systems Theory

## Hypothesis:

The code generated by Power BI to perform ETL operations does not respect DvT and AvT, and thus results in a non-evolvable system.

The visualizations of data widgets in Power BI does not respect SoC, and thus results in a non-evolvable system.



# Experiment 1

## A Basic ETL Data Cleanup Activity

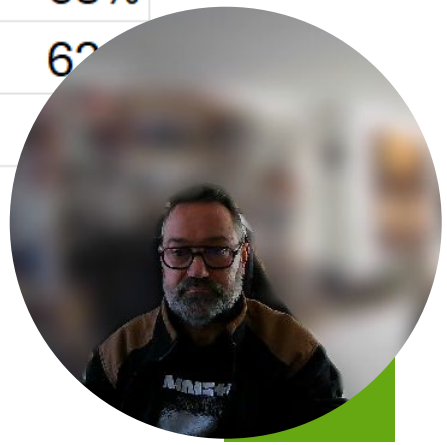
- Read xls file containing tabular data.
- Convert all read data to the “text data type”
- Replace all “empty” and “null” data by a string with value “**None**”

**CHANGE:**

- Add a new column to the source data
  - Apply the same ETL Data Cleanup Activity
  - Check if the same result is obtained.
- Change the name of a column in the data source
  - Apply the same ETL Data Cleanup Activity
  - Check the result

# Experiment 1

| Name | Description | Business Criticality | CAPEX (2023) | Lifecycle: Active | Completion |
|---|---|---|---|---|---|
| ABC123 | some random text 1 | businessOperational | | 2020-01-01 | 87% |
| ABC124 | some random text 2 | businessCritical | | 2020-01-01 | 100% |
| ABC125 | some random text 3 | businessOperational | | | 68% |
| ABC126 | some random text 4 | businessOperational | 100 | 2020-01-01 | 62% |
| ABC127 | some random text 5 | missionCritical | | 2020-01-01 | 62% |
| ABC128 | some random text 6 | businessOperational | 50 | 2022-01-01 | 68% |
| ABC129 | some random text 7 | | | 2025-03-13 | 100% |
| ABC130 | some random text 8 | businessOperational | 0 | 2020-01-01 | 62% |
| ABC131 | some random text 9 | businessOperational | | 2022-01-01 | 68% |
| ABC132 | some random text 10 | businessOperational | | 2020-01-01 | 62 |
| ABC133 | some random text 11 | administrativeService | 3,3 | 2022-01-01 | |
| ABC134 | some random text 12 | businessOperational | 20 | 2022-01-01 | |



# Experiment 1

Transform | Add Column | View | Tools | Help

Data source settings | Manage Parameters | Export query results | Refresh Preview | Properties | Advanced Editor | Manage | Choose Columns | Remove Columns | Keep Rows | Remove Rows | Split Column | Group By | Data Type: Text | Use First Row as Headers | Replace Values | Merge Queries | Append Queries | Combine Files

Data Sources | Parameters | Output Data | Query | Manage Columns | Reduce Rows | Sort | Transform | Combine

```
= Table.ReplaceValue(#"Step 4: Replaced Value null","","**None**",Replacer.ReplaceValue,{"Name", "Description", "Business
```

| | Name | Description | Business Criticality | CAPEX (2023) | Lifecycle: Active | Completion |
|---|---|---|---|---|---|---|
| 1 | ABC123 | some random text 1 | businessOperational | **None** | 01/01/2020 | 0,87 |
| 2 | ABC124 | some random text 2 | businessCritical | **None** | 01/01/2020 | 1 |
| 3 | ABC125 | some random text 3 | businessOperational | **None** | **None** | 0,68 |
| 4 | ABC126 | some random text 4 | businessOperational | 100 | 01/01/2020 | 0,62 |
| 5 | ABC127 | some random text 5 | missionCritical | **None** | 01/01/2020 | 0,62 |
| 6 | ABC128 | some random text 6 | businessOperational | 50 | 01/01/2022 | 0,68 |
| 7 | ABC129 | some random text 7 | **None** | **None** | 13/03/2025 | 1 |
| 8 | ABC130 | some random text 8 | businessOperational | 0 | 01/01/2020 | 0,62 |
| 9 | ABC131 | some random text 9 | businessOperational | **None** | 01/01/2022 | 0,68 |
| 10 | ABC132 | some random text 10 | businessOperational | **None** | 01/01/2020 | 0,62 |
| 11 | ABC133 | some random text 11 | administrativeService | 3,3 | 01/01/2022 | 0,68 |
| 12 | ABC134 | some random text 12 | businessOperational | 20 | 01/01/2022 | 0,68 |

Query Settings

PROPERTIES

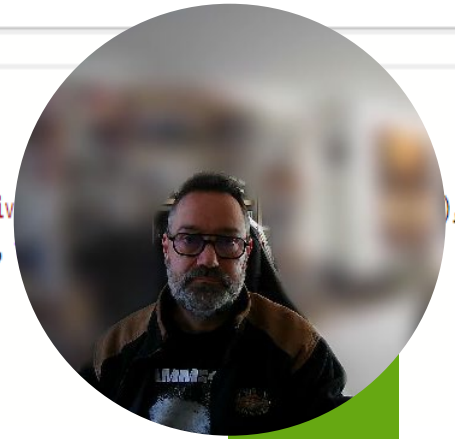
Name: Source

All Properties

APPLIED STEPS

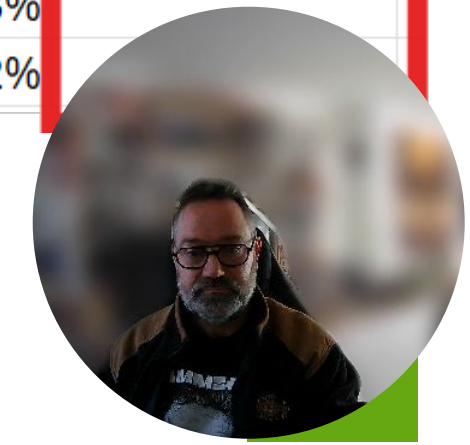
- Source
- Navigation
- Step 2: Promoted Headers
- Step 3: Auto Changed Type
- Step 3: Changed Type txt
- Step 4: Replaced Value null
- Step5: Replaced Value empty string

```
let
    Source = Excel.Workbook(File.Contents("C:\Users\ghaer\Downloads\Source.xlsx"), null, true),
    #"Sheet 1_Sheet" = Source{[Item="Sheet 1",Kind="Sheet"]}[Data],
    #"Step 2: Promoted Headers" = Table.PromoteHeaders(#"Sheet 1_Sheet", [PromoteAllScalars=true]),
    #"Step 3: Changed Type" = Table.TransformColumnTypes(#"Step 2: Promoted Headers",{{"ID", type text}, {"Name", type text}, {"Business Criticality", type text}, {"CAPEX  (2023)", Int64.Type}, {"Lifecycle: Activ
    #"Step 3: Changed Type1" = Table.TransformColumnTypes(#"Step 3: Changed Type",{{"ID", type text}, {"Name", type text}, {"Business Criticality", type text}, {"CAPEX  (2023)", type text}, {"Lifecycle: Active",
    #"Step 4: Replaced Value" = Table.ReplaceValue(#"Step 3: Changed Type1","","**None**",Replacer.ReplaceValue,{"ID", "Name", "Business Criticality", "CAPEX  (2023)", "Lifecycle: Active", "Completion"}),
    #"Step 5: Replaced Value1" = Table.ReplaceValue(#"Step 4: Replaced Value",null,"**None**",Replacer.ReplaceValue,{"ID", "Name", "Business Criticality", "CAPEX  (2023)", "Lifecycle: Active", "Completion"})
in
    #"Step 5: Replaced Value1"
```



# Experiment 1

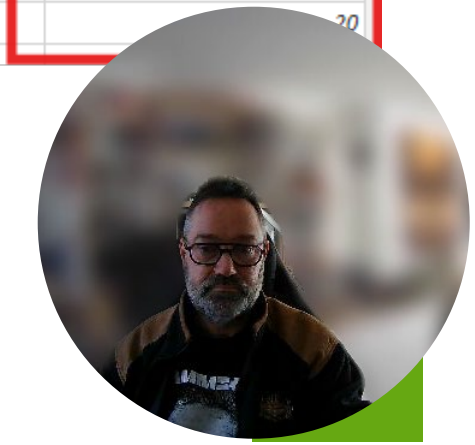
| Name | Description | Business Criticality | CAPEX (2023) | Lifecycle: Active | Completion | CAPEX (2023) 2 |
|---|---|---|---|---|---|---|
| ABC123 | some random text 1 | businessOperational | | 2020-01-01 | 87% | |
| ABC124 | some random text 2 | businessCritical | | 2020-01-01 | 100% | |
| ABC125 | some random text 3 | businessOperational | | | 68% | |
| ABC126 | some random text 4 | businessOperational | 100 | 2020-01-01 | 62% | 100 |
| ABC127 | some random text 5 | missionCritical | | 2020-01-01 | 62% | |
| ABC128 | some random text 6 | businessOperational | 50 | 2022-01-01 | 68% | 50 |
| ABC129 | some random text 7 | | | 2025-03-13 | 100% | |
| ABC130 | some random text 8 | businessOperational | 0 | 2020-01-01 | 62% | 0 |
| ABC131 | some random text 9 | businessOperational | | 2022-01-01 | 68% | |
| ABC132 | some random text 10 | businessOperational | | 2020-01-01 | 62% | |



# Experiment 1

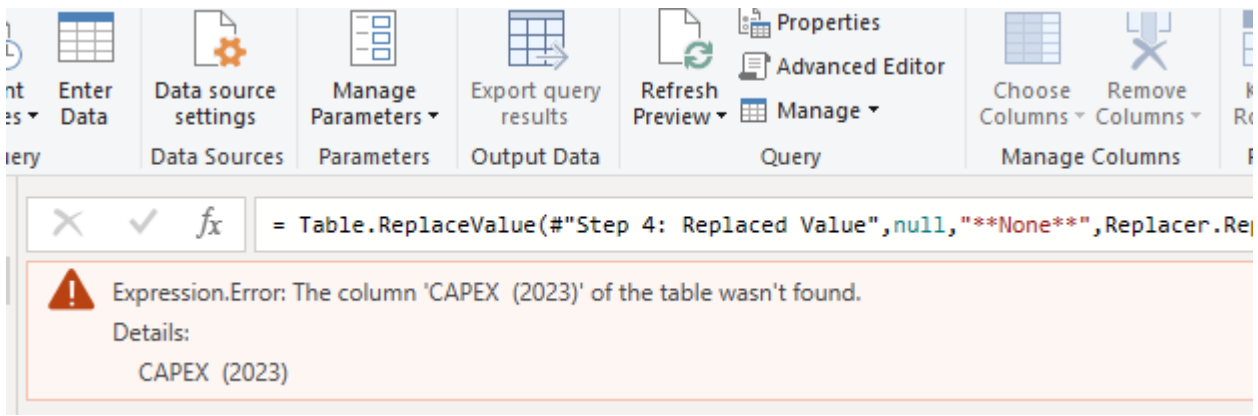
```
= Table.ReplaceValue(#"Step 4: Replaced Value null","","**None**",Replacer.ReplaceValue,{"Name", "Description", "Business Criticality", "CAPEX (2023)",
```

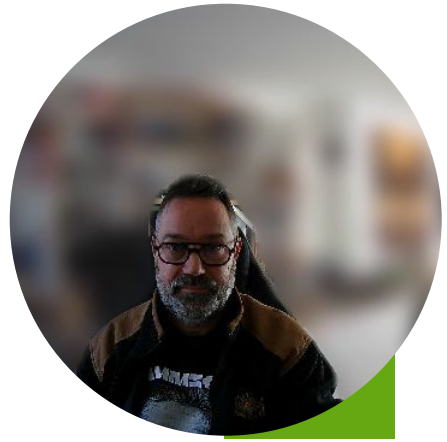
| | Name | Description | Business Criticality | CAPEX (2023) | Lifecycle: Active | Completion | CAPEX (2023) 2 |
|---|---|---|---|---|---|---|---|
| 1 | ABC123 | some random text 1 | businessOperational | **None** | 01/01/2020 | 0,87 | |
| 2 | ABC124 | some random text 2 | businessCritical | **None** | 01/01/2020 | 1 | |
| 3 | ABC125 | some random text 3 | businessOperational | **None** | **None** | 0,68 | |
| 4 | ABC126 | some random text 4 | businessOperational | 100 | 01/01/2020 | 0,62 | *100* |
| 5 | ABC127 | some random text 5 | missionCritical | **None** | 01/01/2020 | 0,62 | |
| 6 | ABC128 | some random text 6 | businessOperational | 50 | 01/01/2022 | 0,68 | *50* |
| 7 | ABC129 | some random text 7 | **None** | **None** | 13/03/2025 | 1 | |
| 8 | ABC130 | some random text 8 | businessOperational | 0 | 01/01/2020 | 0,62 | *0* |
| 9 | ABC131 | some random text 9 | businessOperational | **None** | 01/01/2022 | 0,68 | |
| 10 | ABC132 | some random text 10 | businessOperational | **None** | 01/01/2020 | 0,62 | |
| 11 | ABC133 | some random text 11 | administrativeService | 3,3 | 01/01/2022 | 0,68 | *3,3* |
| 12 | ABC134 | some random text 12 | businessOperational | 20 | 01/01/2022 | 0,68 | *20* |



# Experiment 1

| Name | Description | Business Criticality | CAPEX (2025) | Lifecycle: Active | Completion | CAPEX (2023) 2 |
|---|---|---|---|---|---|---|
| ABC123 | some random text 1 | businessOperational | | 2020-01-01 | 87% | |
| ABC124 | some random text 2 | businessCritical | | 2020-01-01 | 100% | |
| ABC125 | some random text 3 | businessOperational | | | 68% | |
| ABC126 | some random text 4 | businessOperational | 100 | 2020-01-01 | 62% | 100 |
| ABC127 | some random text 5 | missionCritical | | 2020-01-01 | 62% | |
| ABC128 | some random text 6 | businessOperational | 50 | 2022-01-01 | 68% | 50 |
| ABC129 | some random text 7 | | | 2025-03-13 | 100% | |
| ABC130 | some random text 8 | businessOperational | 0 | 2020-01-01 | 62% | 0 |
| ABC131 | some random text 9 | businessOperational | | 2022-01-01 | 68% | |
| ABC132 | some random text 10 | businessOperational | | 2020-01-01 | 62% | |
| ABC133 | some random text 11 | administrativeService | 3,3 | 2022-01-01 | 68% | 3,3 |



# Experiment 1

Experiment outcome:

The ETL steps are not DvT.

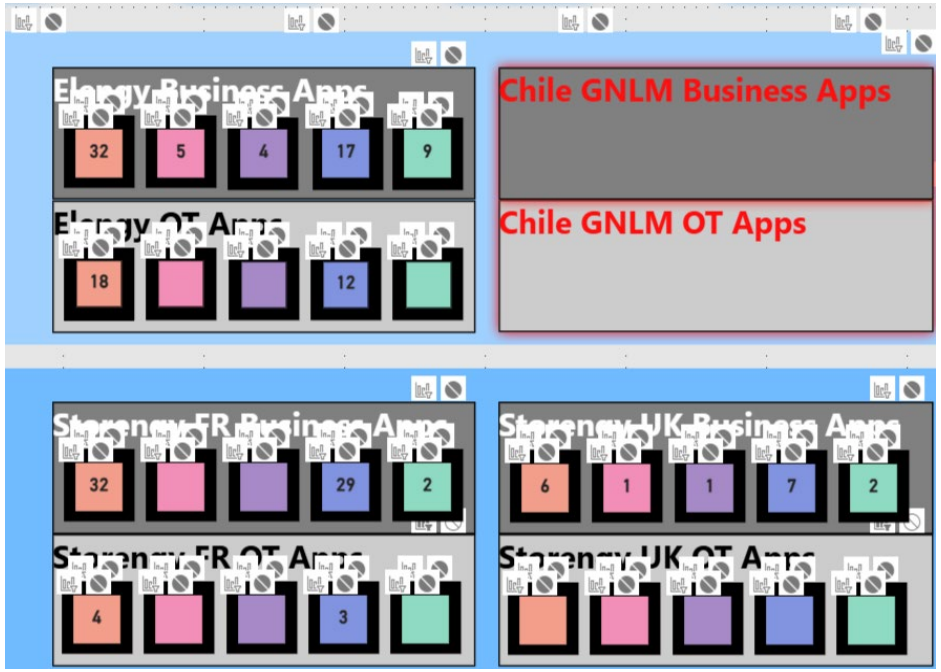
Why?

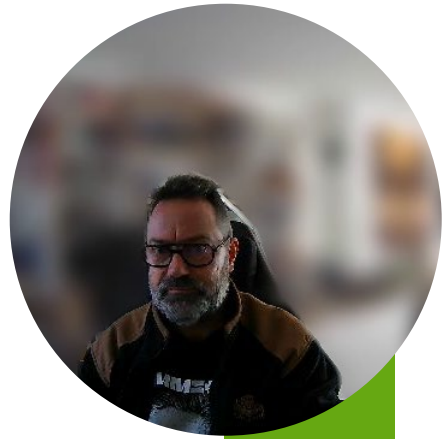
Data Coupling in the Interface

```
let
    Source = Excel.Workbook(File.Contents("C:\Users\ghaer\Downloads\Source.xlsx"), null, true),
    #"Sheet 1_Sheet" = Source{[Item="Sheet 1",Kind="Sheet"]}[Data],
    #"Step 2: Promoted Headers" = Table.PromoteHeaders(#"Sheet 1_Sheet", [PromoteAllScalars=true]),
    #"Step 3: Changed Type" = Table.TransformColumnTypes(#"Step 2: Promoted Headers",{{"ID", type text}, {"Name", type text}, {"Business Criticality", type text}, {"CAPEX  (2023)", Int64.Type}, {"Lifecycle: Activ
    #"Step 3: Changed Type1" = Table.TransformColumnTypes(#"Step 3: Changed Type",{{"ID", type text}, {"Name", type text}, {"Business Criticality", type text}, {"CAPEX  (2023)", type text}, {"Lifecycle: Active",
    #"Step 4: Replaced Value" = Table.ReplaceValue(#"Step 3: Changed Type1","","**None**",Replacer.ReplaceValue,{"ID", "Name", "Business Criticality", "CAPEX  (2023)", "Lifecycle: Active", "Completion"}),
    #"Step 5: Replaced Value1" = Table.ReplaceValue(#"Step 4: Replaced Value",null,"**None**",Replacer.ReplaceValue,{"ID", "Name", "Business Criticality", "CAPEX  (2023)", "Lifecycle: Active", "Completion"})
in
    #"Step 5: Replaced Value1"
```

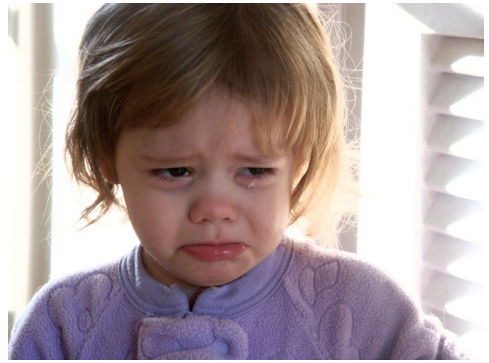
# Experiment 2

Adding a new visual to the Dashboard Canvas.
Visuals interact with each other BY DEFAULT.
The more Visuals you have, the bigger the effort to configure them.
This is a CE, caused by  violating SoC.



# Discussion



These were simple cases!

The Code generate is not Evolvable.

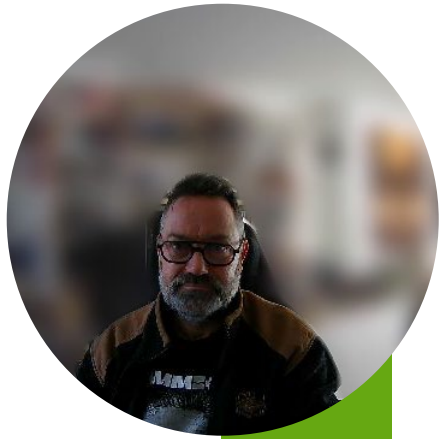
But could have been made Evolvable.

If you want to solve the issue: you need to code.

But it is supposed to be a low-code tool.

The Visual Interactions are not Evolvable.

But could have been made Evolvable.



# Conclusion

## Hypothesis PROVEN:

The code generated by Power BI to perform ETL operations does not respect DvT and AvT, and thus results in a non-evolvable system.

The visualizations of data widgets in Power BI does not respect SoC, and thus results in a non-evolvable system.

What's the point of Low-Code/No-Code if you need to co
it Evolvable?