|  |
| --- |
| Schneider Electric |
| Generating Marimeko |
| How-To ? |

|  |
| --- |
| Gilbert BRAULT  24/02/2017 |

Original design of the Marimekko carried by Robert Mundigl from [www.clearlyandsimply.com](http://www.clearlyandsimply.com)

PowerPoint generation from gitbrent <https://github.com/gitbrent/PptxGenJS>

+ many other open source software

Table of content

[Context 2](#_Toc475795185)

[The workflow encompasses those steps 2](#_Toc475795186)

[The excel Workbook template 3](#_Toc475795187)

[User input in the excel Workbook 3](#_Toc475795188)

[The Dashboard Sheet 3](#_Toc475795189)

[The Colors Sheet 4](#_Toc475795190)

[Generating PowerPoint Marimekko 5](#_Toc475795191)

[Usual workflow 5](#_Toc475795192)

[“File to Render” Panel 5](#_Toc475795193)

[Hide/show json file Editor 5](#_Toc475795194)

[Input File to Render 6](#_Toc475795195)

[Load key 6](#_Toc475795196)

[Encode 6](#_Toc475795197)

[Decode 6](#_Toc475795198)

[Generate Key 6](#_Toc475795199)

[Download File 6](#_Toc475795200)

[“File to execute” Panel 6](#_Toc475795201)

[Hide/show JavaScript Editor 6](#_Toc475795202)

[Input File to execute 6](#_Toc475795203)

[Execute code 6](#_Toc475795204)

[Ciphering marimekko source file 7](#_Toc475795205)

[Ciphering workflow 7](#_Toc475795206)

[Deciphering workflow 7](#_Toc475795207)

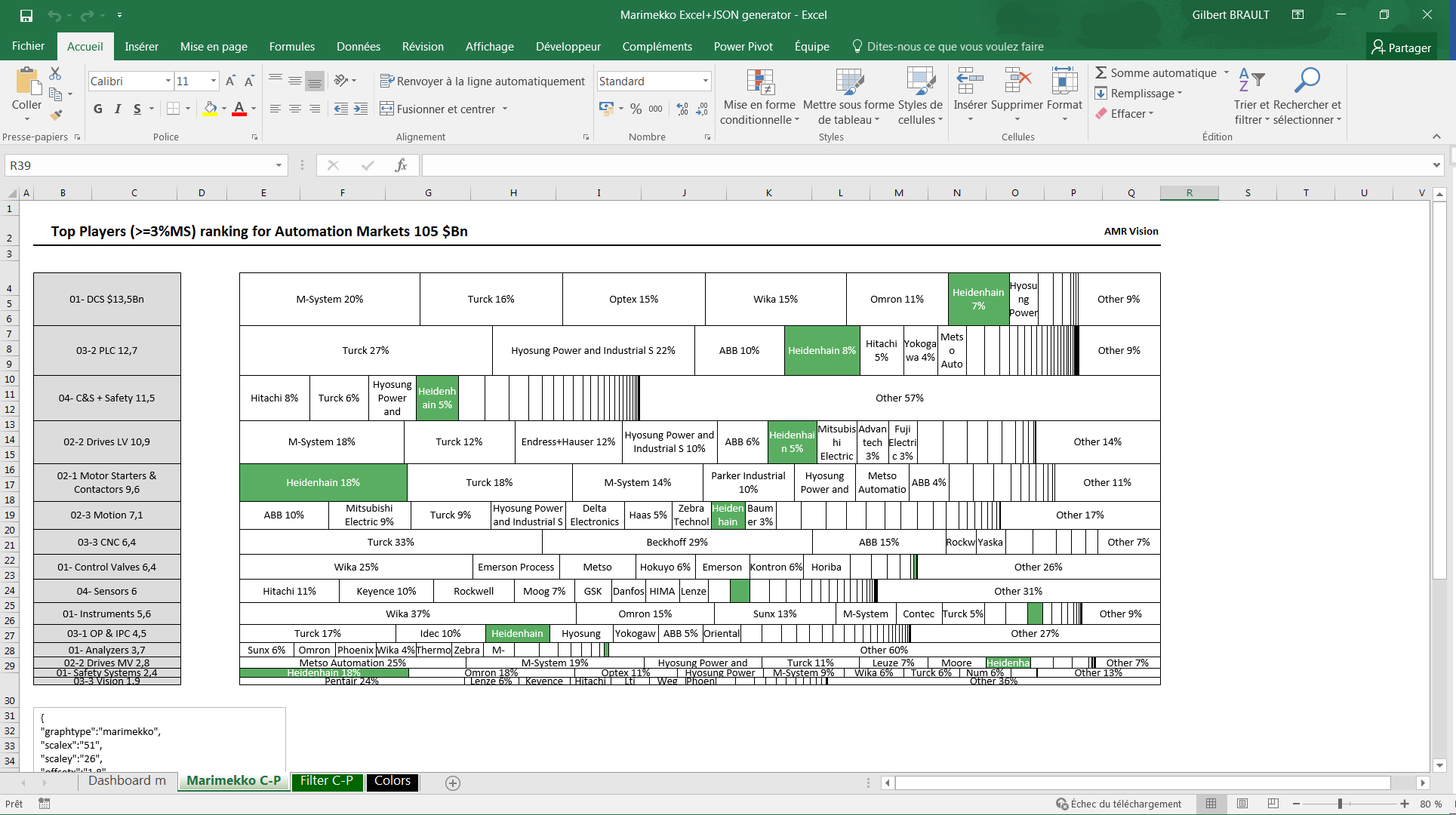
# Context

In many cases, business data comes as a table where columns are for example product lines and rows are competitors, but other settings might be handled, the semantic of rows and column is not fixed by the present tools.

Even if rows and column headers might not actually be competitors and products, we will refer as such in the rest of the document for sake of understanding.



From this table, one wants to have an overview allowing to seize the market at a glance.



This document is presenting the tools providing such kind of graphs from table primary data.

* Input data is expected as an excel double entry table: columns are products, rows, competitors
* Output is an excel graph or a PowerPoint slide graph

## The workflow encompasses those steps

* Get the excel template
* Taylor it to a particular user situation
  + Load double entry table data for each marimekko
  + Set the dashboard marimekko(s) definition and create one Sheet per marimekko output
  + The Colors sheet which defines marimekko expansion parameters
* Run the excel macro generating
  + The excel marimekko
  + The json file defining a marimekko
* To generate the PowerPoint slide
  + Browse to the businessgraphs application:   
    <https://rawgit.com/gbrault/businessgraphs/master/index.html>
  + Load the file(s) generated by the excel macro (naming as parameter, reside(s) in the directory of the workbook) and execute the JavaScript macro (preloaded) => this upload a PowerPoint file complying with user set parameters (see further on for definition)

# The excel Workbook template

The “template” workbook is composed of 4 sheets

* Dashboard m
* Marimekko C-P
* Filter C-P
* Colors

It’s a macro enabled file (.xlsm), user should put appropriate macro settings in excel to run the macros, having the following modules

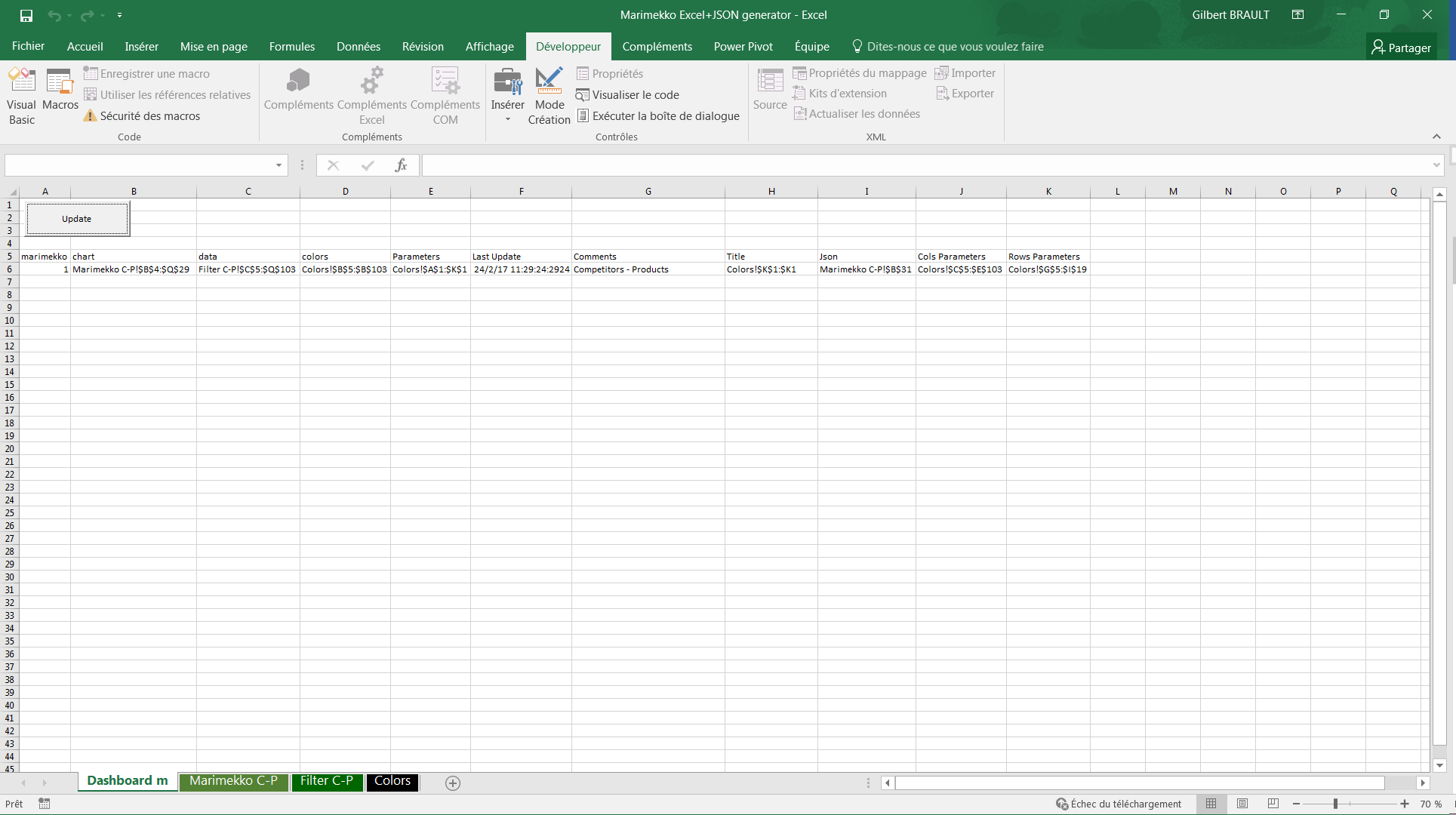
* Dashboard m as a macro when user click on the update button (Update\_Click)
* Tow modules
  + modMarimekko which process the table and produce the excel graph and json marimekko file
  + utils providing functions for the previous module
* Original design of the Marimekko was carried by Robert Mundigl from [www.clearlyandsimply.com](http://www.clearlyandsimply.com)

## User input in the excel Workbook

* The “Dashboard m” is where the various marimekkos are defined
* The “Colors” Sheet is where parameters are set for each defined marimekko
* The “Filter C-P” Sheet is where the double entry table is: on per marimekko

### The Dashboard Sheet

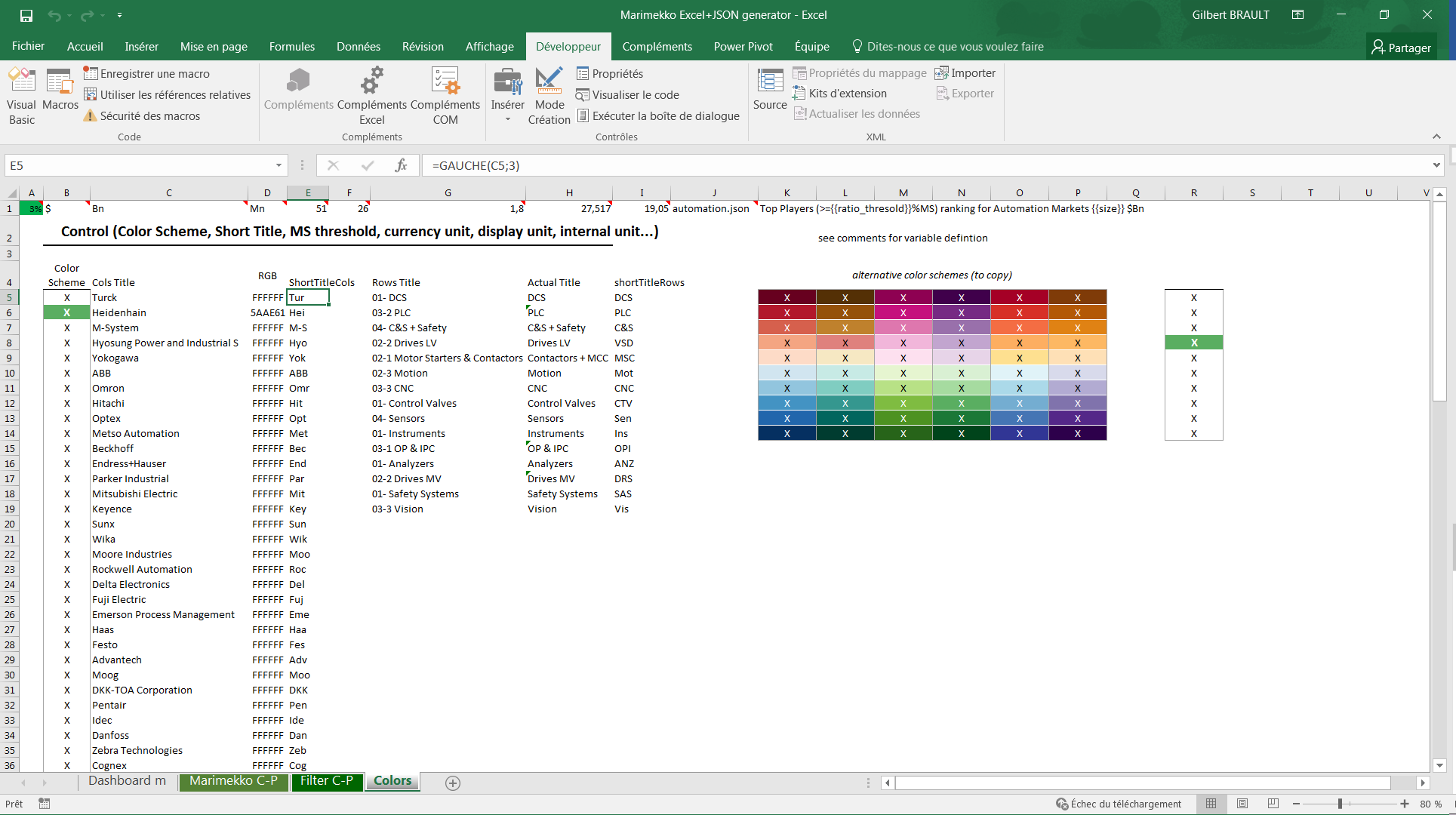
The definition of the marimekkos is given in this sheet. Marimekko definition start in line 6 column 1. The columns are defined as follow



1. The sequence number: from 1 to 20 at most incremented by 1 in natural order
2. Chart: where the marimekko chart is located
3. Data: where is the double entry table, addresses locate numbers data (not including columns or rows headers)
4. Colors: definition of filling colors for each row of the data table
5. Parameters: a set of parameters which control marimekko expansion
6. Last Update: when did the marimekko was calculated
7. Comments: free form cell use for documentation purpose
8. Title: the title line for the json marimekko file
9. Json: where will the json text be displayed
10. Cols Parameters: various parameters which control json graph columns
11. Rows parameters: same as above for rows.

### The Colors Sheet

Usually, on set all parameters, including colors in one sheet



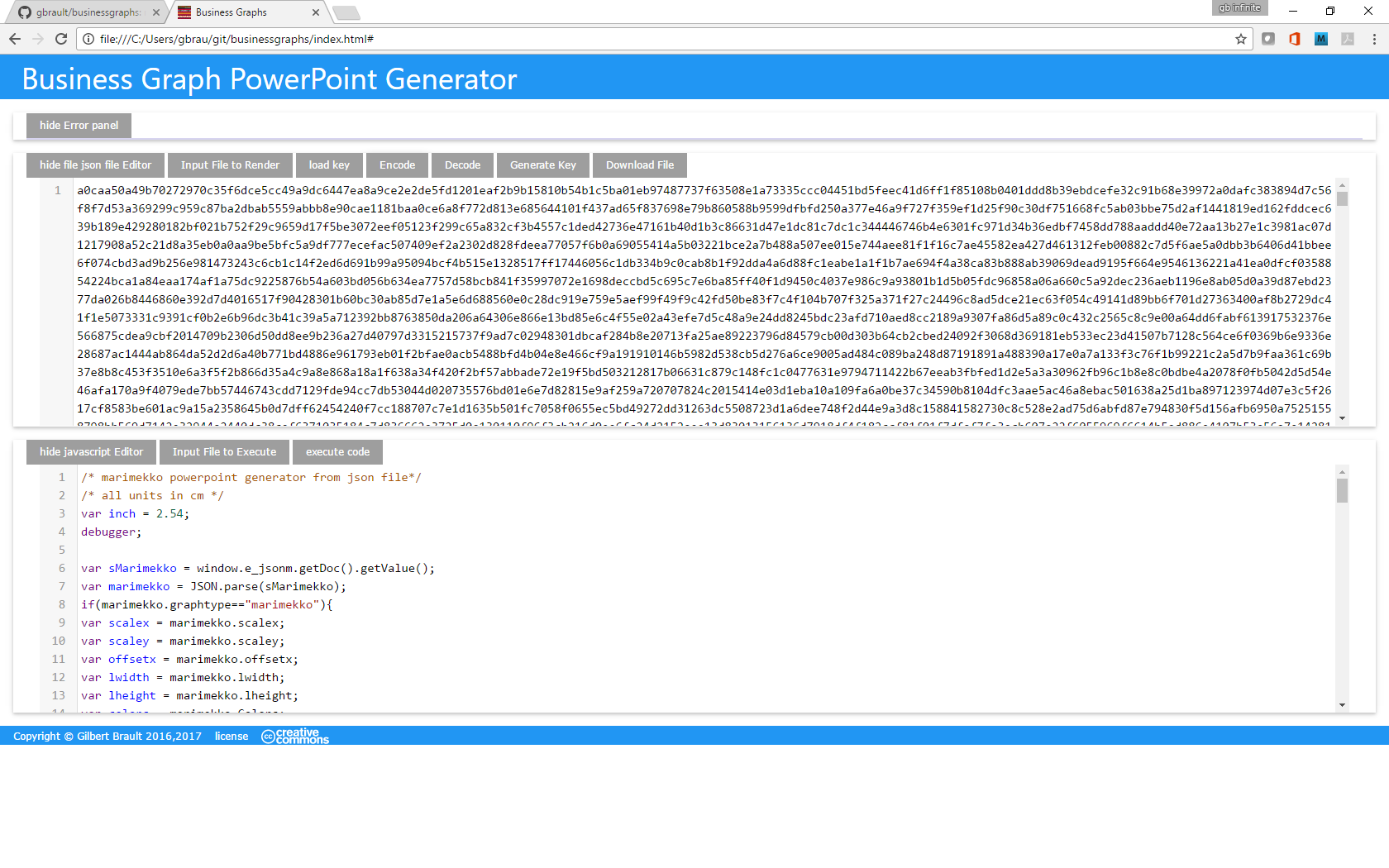
* Parameters: Colors!$A$1:$K$1
  + A1 = Market Share threshold (under that competitor will be attributed to other) {{ratio\_thresold}}
  + B1 = Currency {{currency}}
  + C1 = display unit (display information is in Bn for example) {{display}}
  + D1 = internal unit (data is in Mn for example)
  + E1 = x scaling from excel points to cm (depends on graph size and paper layout)
  + F1 = y scaling from excel points to cm
  + G1 = x offset left for data cells (for the json file)
  + H1 = json slide width
  + I1 = json slide height
* Json: Marimekko C-P!$B$31
  + J1 = file name for the json format
* Title: Colors!$K$1:$K1
  + K1 = title example: “Top Players (>={{ratio\_thresold}}%MS) ranking for Automation Markets {{currency}}{{size}} {{display}}”. Values are substituted.
* Colors: Colors!$B$5:$B$103
  + This column (one row per competitor) should be formatted using template from $K$5:$P:$14 or $R$5:$R$14 (reproduce layout)
* Cols Parameters: Colors!$C$5:$E$103
  + This table is controlling json columns: one entry per competitor
  + Cols Title: $C5:103 the names of competitors as per data rows header
  + It is used to build the Color table $D5:103 in the json file and the short name $E5:103
  + The color is set to getRGB1(B5) (getRGB1 is a macro in utils module)
* Rows Parameters: Colors!$G$5:$I$19
  + This table is controlling json rows: one entry per product line
  + Rows Title: the header name in data columns
  + Actual Title: what name will be displayed
  + shortTitleRows: in case of space constraints shorter name for product

# Generating PowerPoint Marimekko

A one-page client application enables generating the PowerPoint slides defined as json files and produced by the excel above macro.

## Usual workflow

* Click on “Input File to Render”: select the file and open
* Click on “Execute Code” (third panel): The PowerPoint slide is uploaded



This application has three panels

* An error panel to guide user through potential errors
* A “File to Render” Panel, where the user input the json file marimekko to render as a PowerPoint slide
* A “File to execute” Panel, where the user selects the JavaScript macro to run (by default marimekko macro)

## “File to Render” Panel

Seven menu buttons are available

* Hide/show json file Editor
* Input File to Render
* Load key
* Encode
* Decode
* Generate Key
* Download File

### Hide/show json file Editor

Just hide or show the json editor content

### Input File to Render

To select and load a file in the editor, replacing the present one

### Load key

Load a ciphering key to encode/decode data

### Encode

Using the key, cipher the present editor (select all, copy to paste it in another editor)

### Decode

Using the key, decipher the json editor content

### Generate Key

Generate and download a new key to cipher the file content. The name of the key is

### Download File

Download the json editor content to the local computer

## “File to execute” Panel

Three menu button are available

* Hide/show JavaScript Editor
* Input File to execute
* Execute code

### Hide/show JavaScript Editor

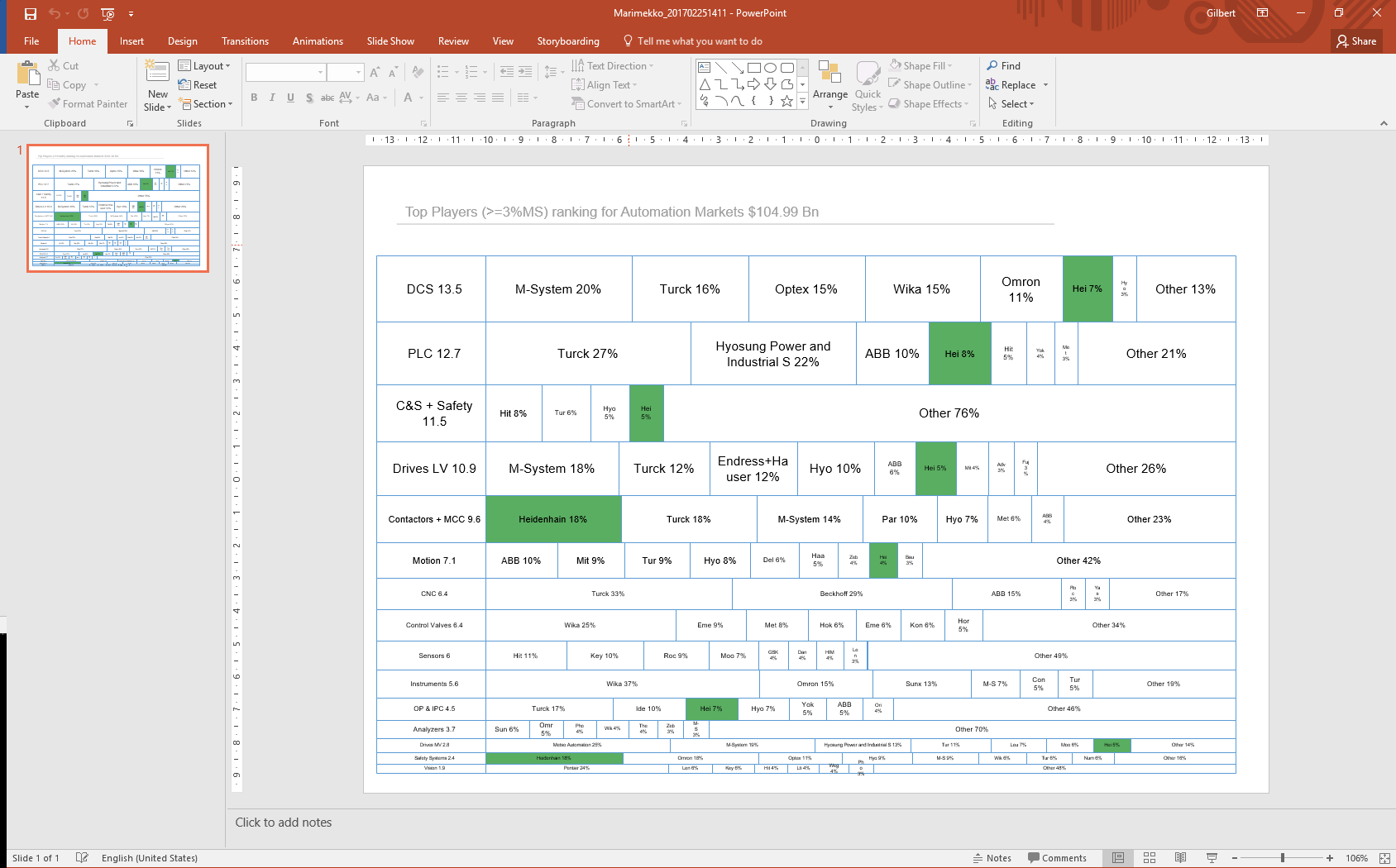
Just hide or show the JavaScript editor content

### Input File to execute

Select and load a JavaScript macro

### Execute code

Execute the macro, which, in case of the marimekko macro, will upload the “PowerPoint” Slide



# Ciphering marimekko source file

If one wants to protect and save the marimekko source file (json file format) this application support ciphering key generation and encoding/decoding of marimekko files (when supporting other file types, the procedure will be the same: we use marimekko here for convenience, it should be the same when new file format will come over time).

## Ciphering workflow

1. A key should be generated: hit the “Generate Key” menu. A file is then downloaded: rename it and save it as this key is going to be the sesame to encode and decode the json file.
2. Load the file you want to cipher using the “Input File to Render” menu
3. Load the key you want to apply using the “load key” menu (select the key file you just generated)
4. Encode using the “Encode” Menu: the content of the json editor is full of hexadecimal number: this is the ciphered version of your file applying the key you have previously loaded
5. Now save the json editor using the “Download File” menu: a file is downloaded in the local file system rename and save it where relevant for you

WARNING: make sure you save the key as it will be needed to decode the file back

## Deciphering workflow

1. Load the ciphered marimekko file “Input File to render” menu
2. Load the key which was used to cypher the above file “load key” menu
3. Decode using the “Decode” menu

The content of the json editor should now show a more human readable content!

WARNING: before using the JavaScript json-marimekko to PowerPoint macro, on needs to decipher the json content.