

# RAC Parking report automation - A user's guide

*mz*

*5 April 2019*

## Contents

<b>1</b>	<b>Intro</b>	<b>1</b>
<b>2</b>	<b>PREREQUISITES</b>	<b>2</b>
<b>3</b>	<b>Data Management Pipeline</b>	<b>3</b>
3.1	Folder structure . . . . .	4
3.2	The pipeline . . . . .	5
<b>4</b>	<b>Instructions for use</b>	<b>5</b>
4.1	Produce a report based on existing data . . . . .	5
4.2	Modify an existing report . . . . .	6
4.3	Preparing a new report based on new data: Wales . . . . .	6
4.4	Preparing a new report based on new data: Scotland . . . . .	7
4.5	Formatting of the reports . . . . .	8
4.6	Misc tips . . . . .	8

## 1 Intro

*Hi everyone! So this is just a draft version of this document, so apologies if it is a bit rambling and rough around the edges, but hopefully you will find it helpful still!*

This document describes the data management pipeline for the parking report automation repository, and describes in detail how users should proceed to add new data and produce new reports. It is highly recommended that you do not skip the first part of the document

But if you do plan on skipping the description of the pipeline, then at least have a look at the diagram on page 3 and note the following:

- The green indicates the files you should be modifying:
  - The **updating-scripts**, which is the main point of entry, where you choose which year you want the report for and add information on any new data required.
  - The **report-rmds**, which are produced once the update script is run, and where you can add/remove/modify the *text* of the report.
- The blue **report-templates** you should be more careful with. They are the general templates from which the report rmds are created. So if there is a typo in there, you might as well fix it there once and for all.

Additionally have a look at the folder structure and note:

- in green the locations of the updating scripts and report rmds, which is where you should be working mainly.
- in orange the `data/01-raw` folder is where all the raw data is kept. Both the existing data I've already saved there, and all the new files you will be adding.
- in pink the `outputs` folder, which holds both the report pdfs as well as all the csv tables that are produced at the same time.

## 2 PREREQUISITES

This was tested on R version 3.5.3, with Rstudio 1.1.463. I do not know how it will behave on older platforms, so maybe just update both.

The following packages need to be installed, and you might as well install them in advance, there is no error checking in the code, so the scripts will probably fail quite unceremoniously.

For your convenience here they are in no particular order to copy/paste:

```
install.packages("tidyr")
install.packages("dplyr")
install.packages("tibble")
install.packages("RefManageR")
install.packages("curl")
install.packages("readxl")
install.packages("tabulizer")
install.packages("kableExtra")
install.packages("showtext")
install.packages("sf")
install.packages("viridis")
install.packages("classInt")
install.packages("colorspace")
install.packages("rmarkdown")
```

Additionally, for you to be able to compile to pdf, you **need to have LaTeX installed**. Now it seems that there is a way to do this simply using TinyTex <https://yihui.name/tinytex/>. I have not tried this, but the guy that wrote it is a legend and I would trust him with my life.

### 3 Data Management Pipeline

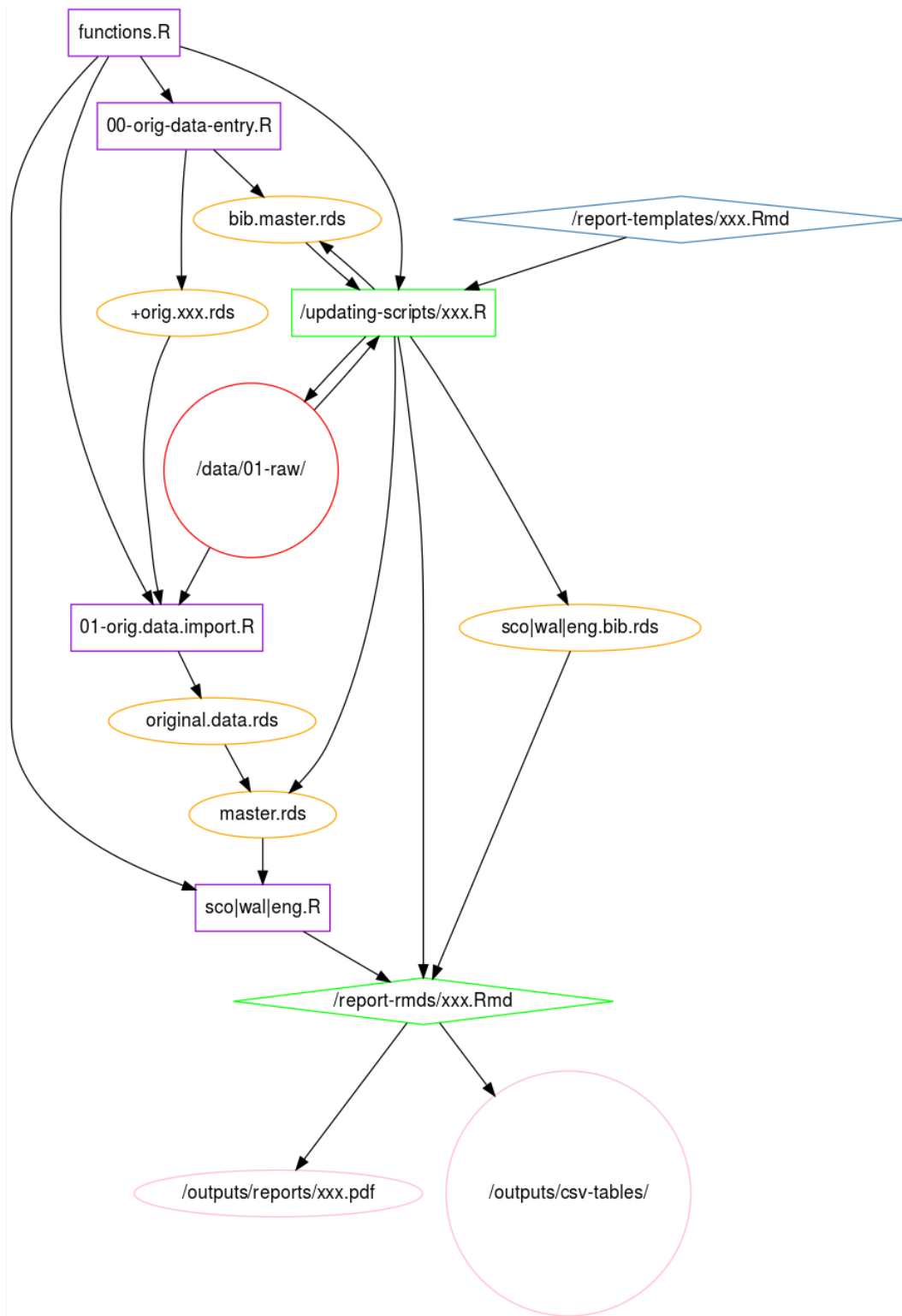


Figure 1: Schematic overview of the data management pipeline

## 3.1 Folder structure

```
|-- code
|   |-- report-rmds
|   |-- report-templates
|   \-- updating-scripts
|-- data
|   |-- 01-raw
|   |   |-- fonts
|   |   \-- maps
|   |-- 02-interim
|   \-- 03-processed
|-- docs
|   |-- admin
|   |-- journal
|   |-- original-reports
|   \-- technical
|       \-- instructions
'-- outputs
.   |-- csv-tables
.   |   |-- scotland-2016-17
.   |   \-- wales-2017-18
.   \-- reports
```

This is a quick overview of what is what in the repository folder structure.

### 3.1.1 Code

Contains all the `.R` and `.Rmd` files. Additionally contains the bibliography files and bib style file, which have to be in the same folder as the Rmds.

#### 3.1.1.1 Top level

- `00-orig-data-entry.R` contains **manually** input data and metadata, including references for existing sources, manually input data from scanned pdfs, and lookup tables for when LA names have spelling variants and typos. *do not modify*

- `01-orig-data-entry.R` contains code to **programmatically** extract and clean data from all the original data sources and produce the `original.data.rds`, which immediately becomes the `master.rds`. *do not modify*
- `functions.R` contains all the helper functions used by pretty much all the other `.R` files. *do not modify*
- `scotland.R` and `wales.R` (and soon to come `england.R`) contain the code run by each `.Rmd` file. This code was originally in the `.Rmd` files themselves, but that made them extremely long and difficult to navigate, so I moved it out. *do not modify*
- `xx-scratchpad.R` - my playground, ignore this, can be deleted in the future.

### 3.1.1.2 updating-scripts

### 3.1.1.3 report-templates

### 3.1.1.4 report-rmds

## 3.2 The pipeline

This section describes the diagram at the start of this document.

## 4 Instructions for use

These instructions are split into three main parts for three main objectives you might have:

1. You want to just try it out: produce a report based on existing data.
2. You want to tweak a report you have already produced.
3. New data has been published and you want to input it and produce a new report. (separate instructions for Wales and Scotland)

### 4.1 Produce a report based on existing data

Although backward compatibility is not in any way guaranteed, it does exist to some extent. So while this pipeline was not meant for you to be able to reproduce old reports, it was a good idea to include that option, since differences in previous years' data inputs made it easier to code against future eventualities.

**I have not tested this for all possible years, but you should be fine producing reports for the last couple of years**

1. Open the `updating-scripts/wales-update-script.R`.
2. Scroll to the MANDATORY INPUTS section.
3. Change the `current.year` field. e.g. to 2016.

4. Make sure `add.new.data` is set to `FALSE`.
5. Make sure `recompile.rmd` is set to `FALSE`.
6. Now run through the script. You can source it as well, but I recommend just Ctrl+Enter-ing through it all, so you will be able to see any errors or messages.
7. When you get to the end, the file should start compiling. It shouldn't take too long.
8. Now go to `outputs/reports/` and have a look at the pdf just created.
9. If you are unhappy with something in the pdf, go to the Section 4.2 describing how you tweak existing reports.

## 4.2 Modify an existing report

You have already produced a report, but you want to tweak it.

1. Open the relevant `..Rmd` file in `code\report-rmds`.
2. Now you can modify the text. Have a look at section `~ref{format}` on formatting for some tips. But the basic rule is to really only change the text portions and avoid changing anything that is in the code chunks, or in inline code. But of course you can, especially captions and table headers etc, just be careful.
3. When you are happy with your modifications, go back to the correct update-script.
4. No, you cannot *knit* the document from the `.Rmd` file itself.<sup>1</sup> Go back to the update-script.
5. Now make sure the following holds:
  - the `current.year` variable is set to the same year as the report you have just modified (check the file name to be sure)
  - Make sure `add.new.data` is set to `FALSE`.
  - Make sure `recompile.rmd` is set to `TRUE`. This is important, otherwise all your tweaking work will be overwritten.
6. Now run the whole script again. It will recompile the pdf with our updates.
7. Have a look at the pdf, and if you are not satisfied go back to point 1.

## 4.3 Preparing a new report based on new data: Wales

1. When the new parking data for Wales becomes available, you need to download the csv files from this website: Road and transport revenue expenditure by authority
2. You will need to export three tables using the export button at the top of the table. Make sure you:
  - **export to .csv, without headers, without metadata**
  - save them to `data/01-raw`
  - pick a sensible name - e.g. the ones suggested below.
3. Select the following three tables:

---

<sup>1</sup>I mean you can, but don't. What will happen is the pdf will get compiled into the wrong folder, and all the temp files and logs will also clutter it up. The idea is for the `report-rmds` folder to just have the `rmds` and the `pdfs` to be in the output folder.

- column: Gross Expenditure; row: Parking of vehicles -> `wal-exp-17-18.csv`
- column: Total Income; row: Parking of vehicles -> `wal-inc-17-18.csv`
- column: Net current cost row: Total transport planning, highways, rows and transport -> `wal-trans-17-18.csv`

4. Now open `wales-update-script.R` and scroll down to the *Mandatory inputs* section.
5. Change the `current.year` field. to the most recent year the data has become available for
6. Make sure `add.new.data` is changed to `TRUE`.
7. Make sure `recompile.rmd` is set to `FALSE`.
8. Then continue to the *Manual input* section.
9. Enter the filenames of the three files you have just downloaded. Just the name, not the path.
10. Update the access date.
11. Now run through the whole file. The script will update the `master.rds` file with the new data as well as updating the bibliography to include the newly added data. Finally it will produce an `.Rmd`, a pdf and the csv tables.
12. If you wish to modify the report go to Section 4.2.

#### 4.4 Preparing a new report based on new data: Scotland

1. When the new parking data for Scotland becomes available, you need to download following files, while *taking note of the urls* where you got them
2. *Income and expenditure* is available on this website: Scottish Local Government Financial Statistics, look for the “Annex A by LA” link. Download the file to `/data/01-raw` e.g. as `sco.i.e.17-18.xlsx`
3. *Abredeen data* is available on this website: Annual accounts. Download the file for this year to `/data/01-raw` e.g. as `sco.abd.17-18.pdf`
4. *PCN data* is available on this website: Transport Scotland. Search for the DPE report for the current year and download it to `/data/01-raw` e.g. as `sco.pcn.17-18.pdf`
5. Now open `scotland-update-script.R` and scroll down to the *Mandatory inputs* section.
6. Change the `current.year` field. to the most recent year the data has become available for
7. Make sure `add.new.data` is changed to `TRUE`.
8. Make sure `recompile.rmd` is set to `FALSE`.
9. Then continue to the *Manual input* section.
10. For each of the three files you will need to manually input all the information requested in the script. This includes bibliographical info like title, link, date accessed for the files.
11. Additionally for the Excel file you will need to open the files and find which cells/sheets the relevant data is on and input that as well.
12. For the Aberdeen data you will need to open the pdf and look for the heading “Trading operations”, usually about 80 pages in, and a few pages further you will find the table with the row “Provision and Management of Car parking facilities”. You will need the turnover and expenditure numbers for the last year, which you input in the script as well.
13. For the DPE/PCN data you need to open the file and check which pages in the pdf the three required tables are located and input them.
14. Now run through the whole file. The script will update the `master.rds` file with the new data as well as updating the bibliography to include the newly added data. Finally it will produce an `.Rmd`, a pdf and the csv tables.
15. If you wish to modify the report go to Section 4.2.

## 4.5 Formatting of the reports

Once you have the pdf there will invariably be things you want to change. Here are some tips as well as some things that you cannot (easily change):

Body text is straightforward to change <sup>2</sup>. But be careful. You should have colour-coded syntax in Rstudio, so you can easily see what is plain text (white) and what is inline code (orange on my computer). Pay attention to this!! Never change anything orange!

In some places where I felt the text will need your attention I have highlighted it in red in the pdf. In the Rmd this will be bracketed by `\color{red}` and `\color{black}`. When you have modified the text to your satisfaction simply delete these tags, and the text will be black again.

- The positioning of the figures

## 4.6 Misc tips

- Be very careful with the YAML headers at the top of the .Rmd files. They are super sensitive to indentation, and if you copy paste them around it will guaranteed get messed up!

---

<sup>2</sup>Here is a link to the rmarkdown cheatsheet with all basic formatting tips. Print it out and stick it on your wall.