IMPROVING THE DATA ANALYSIS CYCLE IN R

BY ALEX DOLPHIN

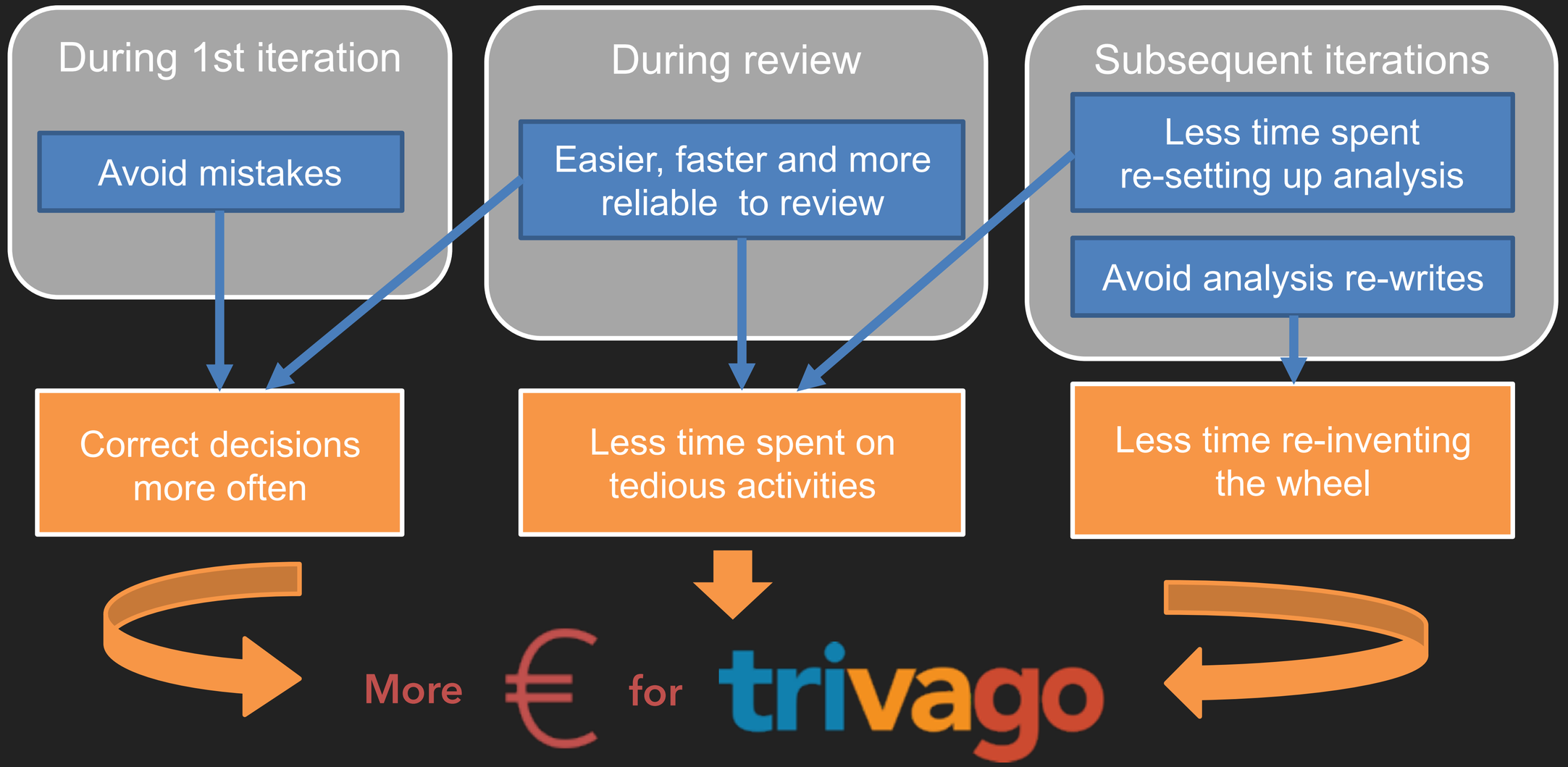
OVERVIEW

- What makes a good analysis?
- Why is it important?
- How do we currently do it?
- How can we do it even better?
- Challenges remaining

WHAT MAKES A GOOD ANALYSIS?

Attribute	Me	Another DS	Non-technical person
Clear code documentation	√	√	
Code understandable	√	√	
Code portable	√	√	
Easy to extend	√	√	√

WHY IS IT IMPORTANT?



HOW DO WE CURRENTLY DO ANALYSES?

Marketplace Intelligence / froggle-libs

Source

develop ··· | froggle-libs / **trv.muffintin** /

Source	Description
..	
trv	
.gitignore	MPI-1010: remove test commmited by mistake
MANIFEST.in	MPI-1010: further comments from Andres and lint
README.md	BUGFIX: Updated README to for uploading packa
setup.cfg	MPI-1010: muffintin in one command
setup.py	MPI-1010: further comments from Andres and lint

README.md

muffintin: trivago's data analysis template

This is a generic template to create reproducible data science analysis at trivago.
It is based on an existing data science template, which uses the python package cookiecutter.

Motivation

The motivations for this template are the following:

- to cut as much as possible the overhead of staring an analysis: for example, creating folder structure, or writing scripts to c
- to make our analyses completely reproducible, by using `make` and explicitely including instructions to install dependencie

The main features are:

- ability to automatically either link to or create a JIRA issue. The analysis folder and report files will also be automatically na
- includes templates for the most commonly used analysis workflows (at the moment only `impaler` + `R`).
- includes `Makefile` template

Installation

Simply run

```
pip install --extra-index-url https://artifactory.tcs.trv.cloud/artifactory/api/pypi/pypi-local/simple trv.muffintin
```

MUFFINTIN

Requirements

- In order to fully reproduce the analysis you will need to have `Docker` in your system.
- You also need to have a `kerberos` keytab file in order to be able to run queries in `Hadoop`

Location of keytab file

The `Makefile` will look for the keytab file in two locations:

1. A *default* location -- `~/$(USER).keytab`
2. Path set in the environment variable `KRB5_KTNAME`

If your keytab is not it one of these locations then you will need to point the `Makefile` accordingly (`find_keytab` target)

User for queries

The `Makefile` also assumes that you intend to run `Hadoop` queries under the user defined by the `USER` environment variable.

If you want to use a different user then you will also need to set up the variable at the beginning of the `Makefile`, e.g.

```
USER = MPIDev
```

Running the analysis

Step 1: setup configuration files to run Hadoop queries under your user

Simply run

```
make setup
```

Step 2: run docker container

To boot up the analysis' Docker container simply execute

```
docker-compose up
```

The first time you execute this it will take some extra time to download and build the image.

Step 3: reproduce the analysis

To regenerate the analysis from scratch you can run:

```
docker-compose exec analysis make
```

The final output should then be created in `reports/bex_api_vs_ftp.html`

Step 4: explore and extend the analysis

The `docker-compose` command from **Step 2** will also launch an instance of Rstudio server, which you can access by pointing yo

From here you can interactively explore and extend the analysis.

DOCKER

{analysis_name}

Makefile

get_data

get_data.R

process_data

process_data.R

analysis

analysis.Rmd

results

.gitkeep

	Muffintin + Docker
Standardised structure	√
Runs out-of-the-box	√
Easy to rerun	?
Easy to review	?
Easy to extend	?

ALL GOOD POINTS, BUT...

WHY BOTHER THE DATA SCIENTIST AGAIN?

WHY WAIT A DAY TO HAVE THE ANALYSIS EXTENDED?

HOW CAN WE FIX THIS?

Bookings_per_hour

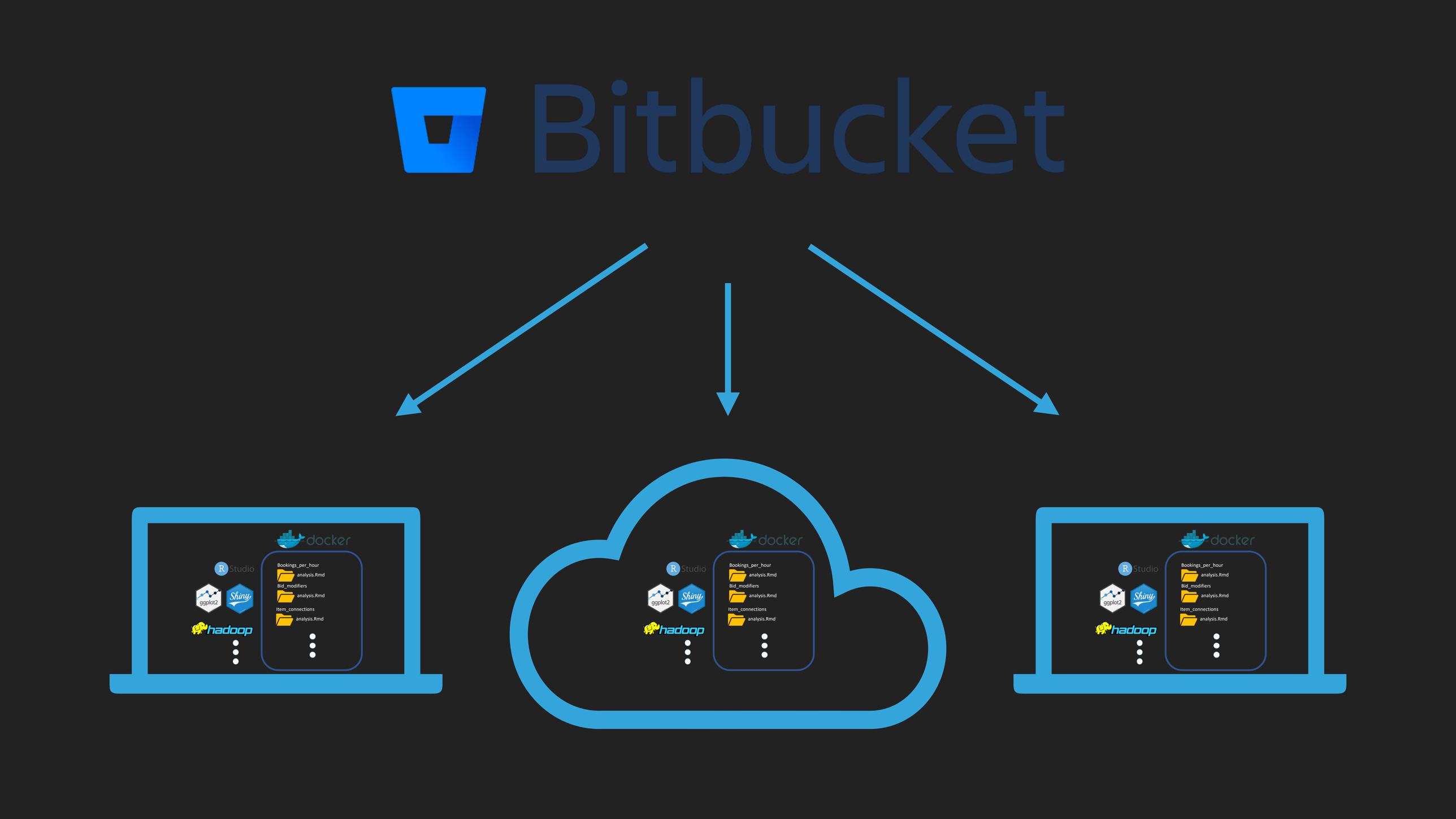
analysis.Rmd

Bid_modifiers

analysis.Rmd

Item_connections

analysis.Rmd


Bitbucket
docker
RStudio
ggplot2
Shiny
hadoop
Bookings_per_hour
analysis.Rmd
Bid_modifiers
analysis.Rmd
Item_connections
analysis.Rmd

HOW DOES THIS IMPROVE THINGS?

~/Documents/git/shiny-analyses/apps/EXAMPLE - master - RStudio

Go to file/function | Addins | EXAMPLE

hourly_bookings.Rmd

```
---
title: "Bookings per hour"
author: "Alex Dolphin"
date: "29/08/2019"
output:
    html_document:
        code_folding: hide
runtime: shiny
---

```{r setup, include=FALSE}
library(magrittr)
library(ggplot2)
library(plotly)
library(chimpala)
library(glue)
library(shiny)
library(data.table)
knitr::opts_chunk$set(echo = TRUE)

lookback_days <- 2
n_top_bookings <- 5
```


Introduction
```

9:4 | Bookings per hour | R Markdown

CAN BE WRITTEN IN ONE R MARKDOWN FILE

Environment | History | Conn

Import Dataset | Global Environment

Data	
bookin…	97887 obs. of …
top_bo…	1348 obs. of 7…
unique…	323 obs. of 2 …
Values	
endDate	"20191016"
lookba…	3
n_top_…	5
startD…	"20191014"
Functions	
gen_na…	function (n)
myFun	function (n = …

Files | Plots | Packages | He

New Folder | Delete | Renam

analyses > apps > EXAMPLE

Name

..

hourly_bookings.Rmd

hourly_bookings.R

Console | Terminal | Jobs

~/Documents/git/shiny-analyses/apps/EXAMPLE/

```
R version 3.6.1 (2019-07-05) -- "Action of the Toes"
Copyright (C) 2019 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin18.6.0 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.
```

LOCAL SHINY SERVER: INSTANT OUTPUT

Screenshot

Bookings per hour

Not Secure | mpi.trv/apps/EXAMPLE/

Apps | Impala | Hive | Hadoop | trv general | BI | Hive UDFs | CK

Select partner IDs

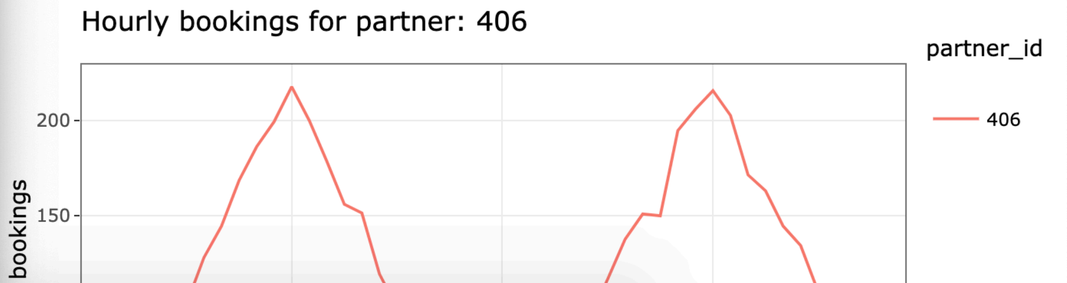
406 - Partner XAGAF7555N

Hide

```
output$bookings_plot <- renderPlotly({
    req(input$selected_partner)
    g <- ggplot(booking_data[partner_id %in% input$selected_partner]) +
        geom_line(aes(x=booking_hour, y=total_bookings, col=partner_id)) +
        theme_bw() +
        labs(
            title=paste0("Hourly bookings for partner: ", input$selected_partner),
            x="Date of booking",
            y="Number of bookings"
        )
    ggplotly(g, dynamicTicks=TRUE)
})

output$booking_amount_plot <- renderPlotly({
    req(input$selected_partner)
    g <- ggplot(booking_data[partner_id %in% input$selected_partner]) +
        geom_line(aes(x=booking_hour, y=total_booking_amount, col=partner_id)) +
        theme_bw() +
        labs(
            title=paste0("Hourly booking amount for partner: ", input$selected_partn
er),
            x="Date of booking",
            y="Total booking amount (euros)"
        )
    ggplotly(g, dynamicTicks=TRUE)
})

plotlyOutput("bookings_plot")
```



Extendible!

One file!

Code blocks: easy review!

Powered by Docker!

Select partner IDs

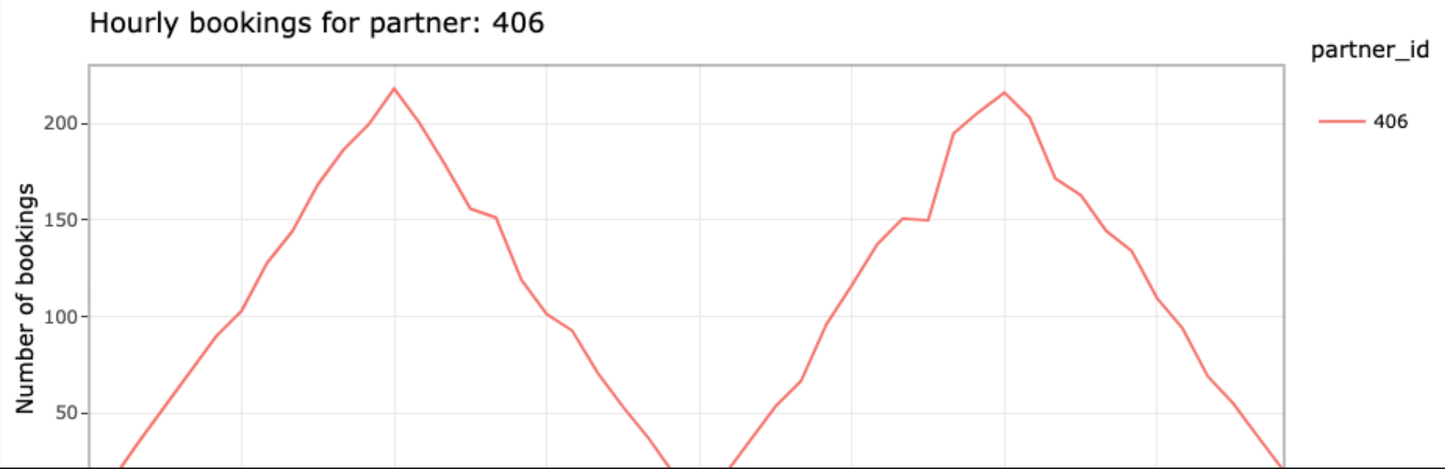
406 - Partner XAGAF7555N

Hide

```
output$bookings_plot <- renderPlotly({
    req(input$selected_partner)
    g <- ggplot(booking_data[partner_id %in% input$selected_partner]) +
        geom_line(aes(x=booking_hour, y=total_bookings, col=partner_id)) +
        theme_bw() +
        labs(
            title=paste0("Hourly bookings for partner: ", input$selected_partner),
            x="Date of booking",
            y="Number of bookings"
        )
    ggplotly(g, dynamicTicks=TRUE)
})

output$booking_amount_plot <- renderPlotly({
    req(input$selected_partner)
    g <- ggplot(booking_data[partner_id %in% input$selected_partner]) +
        geom_line(aes(x=booking_hour, y=total_booking_amount, col=partner_id)) +
        theme_bw() +
        labs(
            title=paste0("Hourly booking amount for partner: ", input$selected_partner),
            x="Date of booking",
            y="Total booking amount (euros)"
        )
    ggplotly(g, dynamicTicks=TRUE)
})

plotlyOutput("bookings_plot")
```



LET'S SEE IT IN ACTION!

- Shiny server repository
- Example analyses repository
- Local server
- MPI server

CHALLENGES REMAINING

- Maintaining a Docker image with all required packages
- Package updates in Docker image could break existing apps
- Caching: don’t let users spam Hadoop
- Date inputs tricky, require data to be constructed reactively
- Continuous deployment of apps to external server: Post-commit hooks?