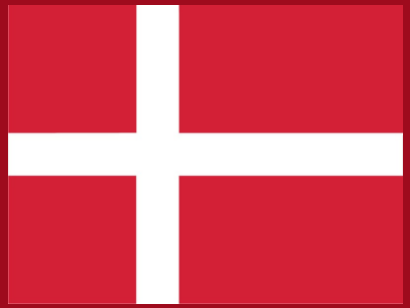
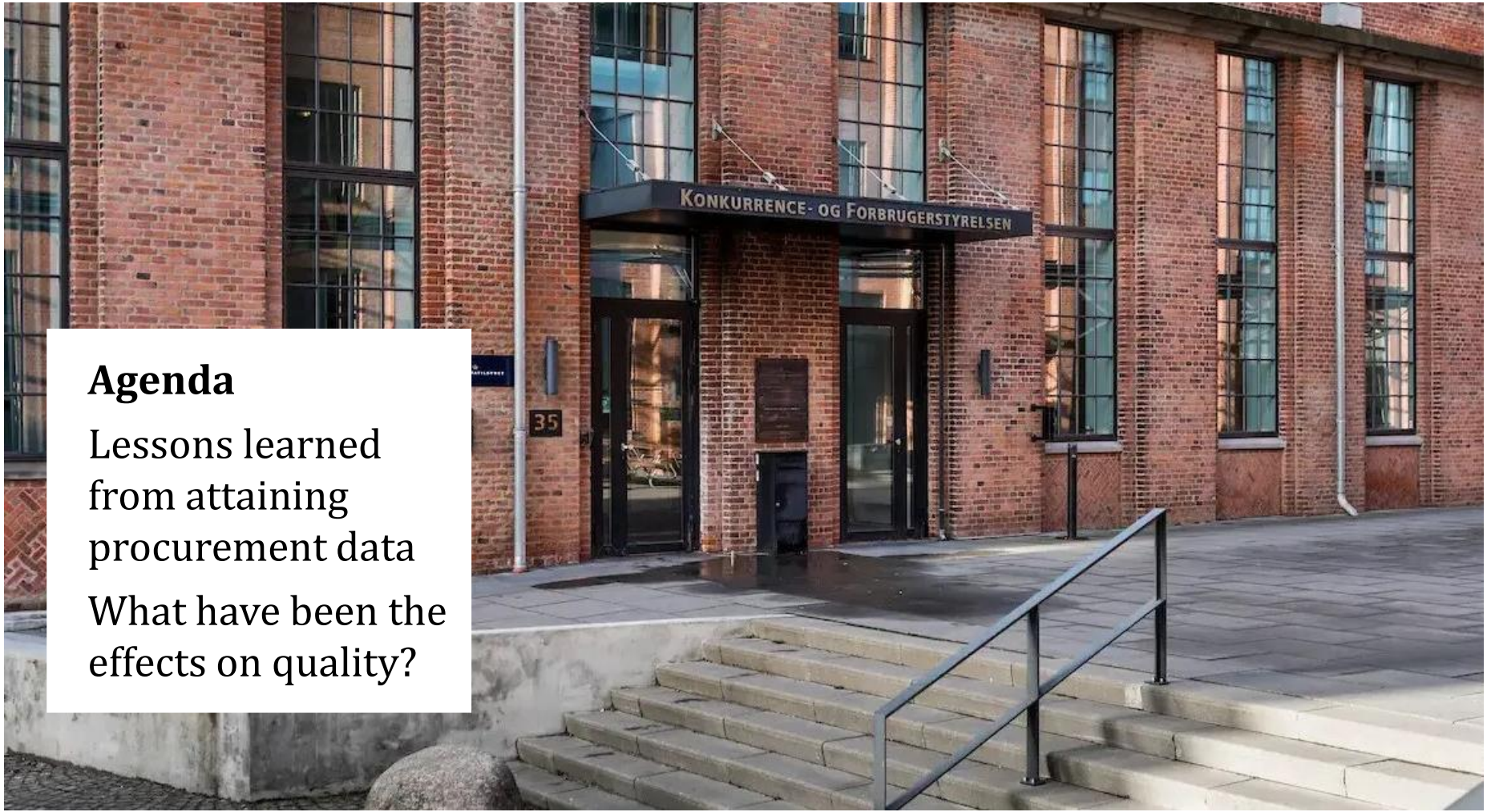
Improving data quality

A Danish case study

Lily Jacobsen Persson – 3. December 2024

Danish Competition and Consumer Authority





Agenda

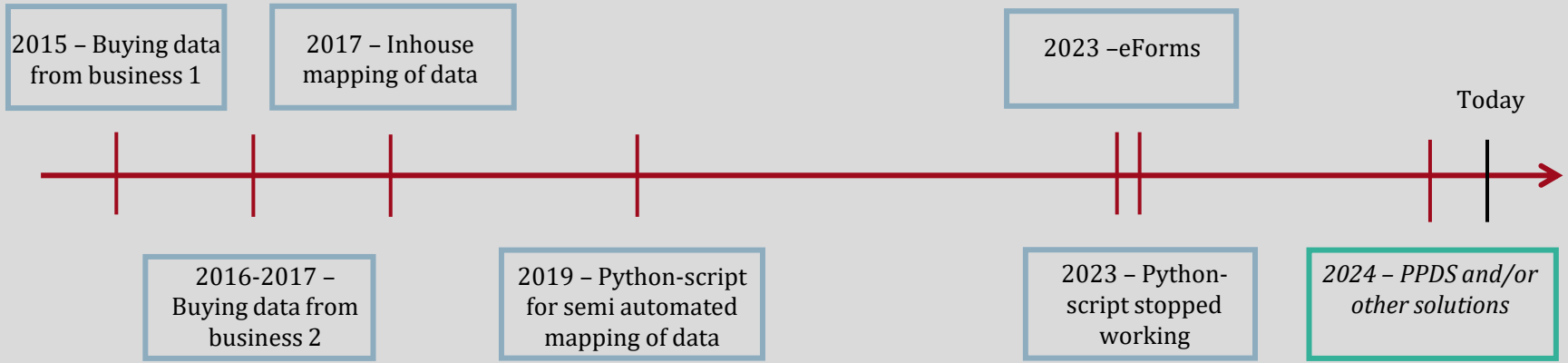
Lessons learned from attaining procurement data

What have been the effects on quality?

Lessons learned from attaining procurement data

Danish Competition and Consumer Authority

Timeline of procurement data



2015 - Buying data from business 1

Purpose and approach

- Measure the impact of the new directive
- Very manual approach centered around CANs and associated documents (the CN, modifications, prior notices)

Lessons learned

- A lot of back and forth with the supplier on getting the desired dataset
- It cost a bit

2017 – Buying data from business 2

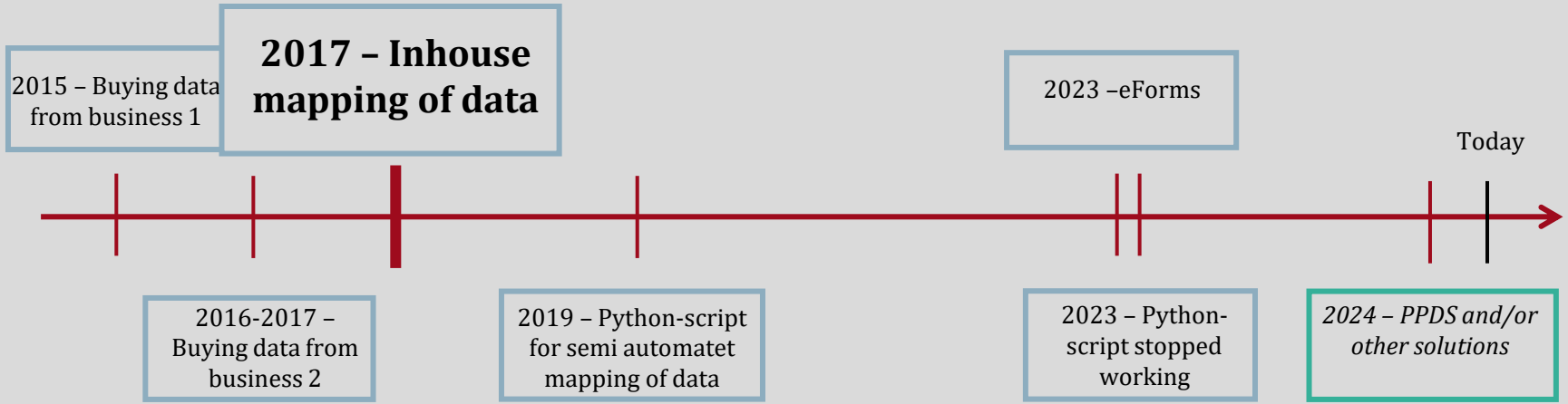
Purpose and approach

- A new company won the contract mapping 2016 and half of 2017
- Different approach with web-scraping TED to obtain data

Lessons learned

- Lower data quality and harder to identify the errors due to scripting errors
- No ongoing information on procurement
- Lack of control over the process and decisions
- Still cost a bit

Timeline of procurement data



2017 – Our own manual mapping

Purpose and approach

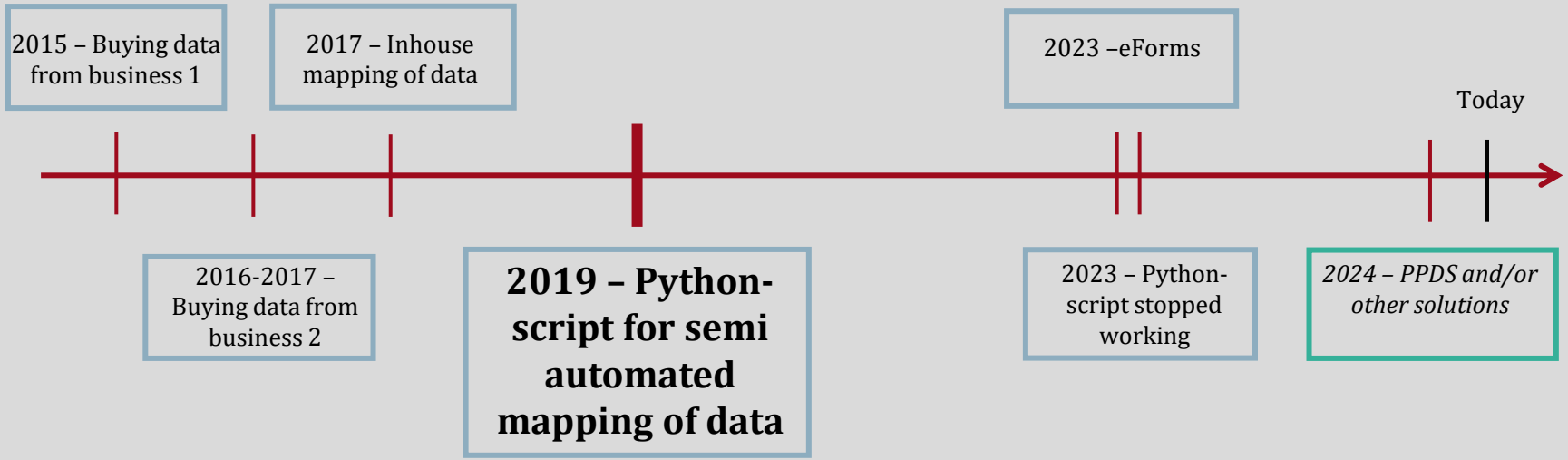
- New in-house unit with 3-4 student assistants
- Ongoing discussions on good practices of mapping data, methodology, practises/reality vs. rules with our legal experts
- Procedures for contacting procuring authorties when identifing missing information or errors
- Procedures for quality assurance every month

2017 – Our own manual mapping

Lessons learned

- Knowledge on what information can be mapped and how to map tenders
- Understanding practises of procuring authorities
- Limitations of data
- Less expensive and other task for student assistants
- Not efficient – and hard to keep the task interesting

Timeline of procurement data

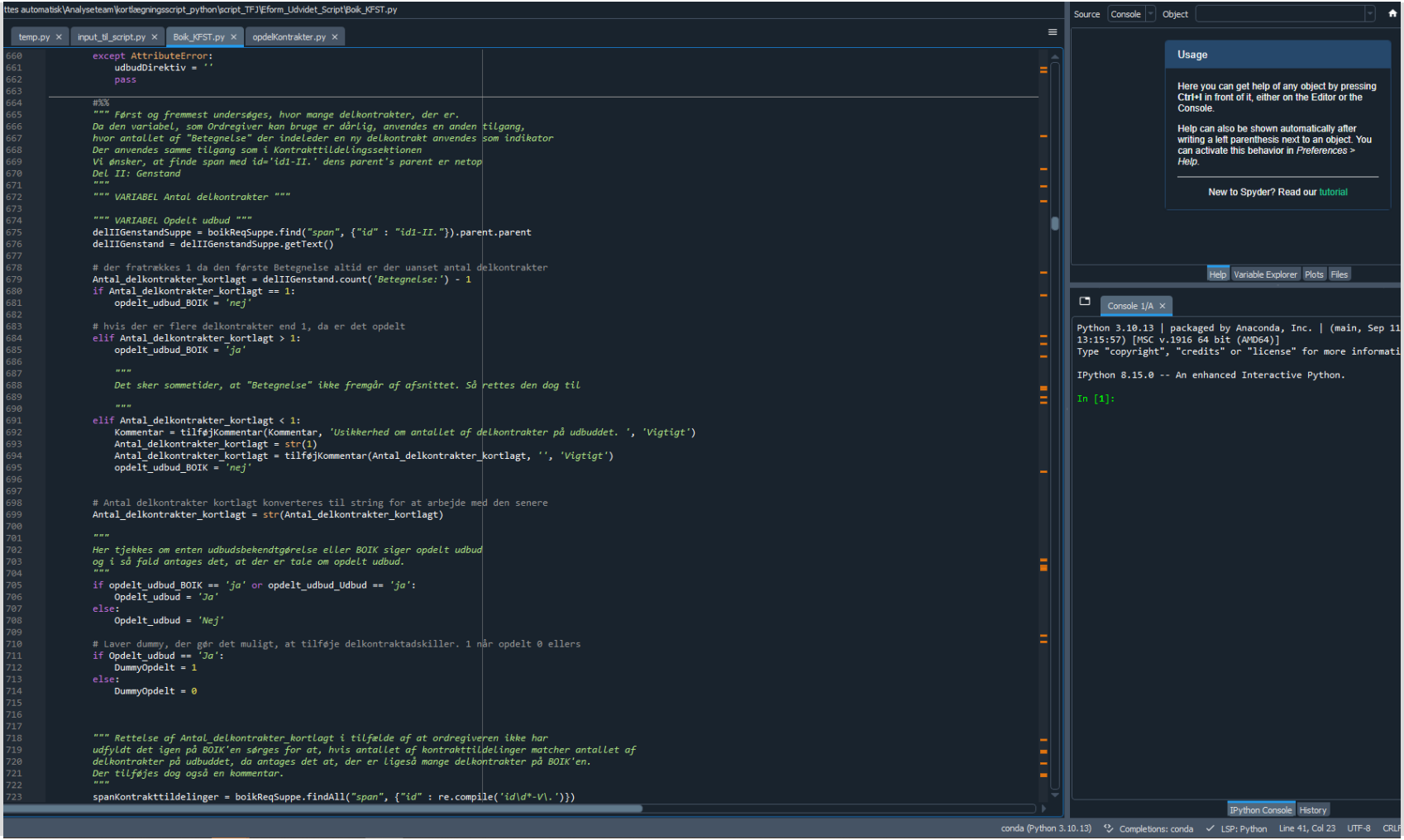


2019 – Automation of mapping data

- We had a well-developed methodology that could be translated to code
- A student assistant with a gift for programming set forth creating a web-scraping script

2019 – Automation of mapping data

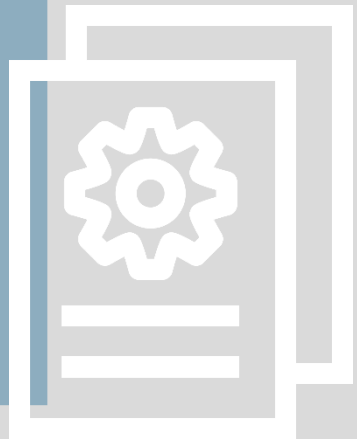
- Built a web-scraping script based on our methodology
- Mapped data from CANs and associated notices
- Corrected information automatically (when possible)
- Highlighted/marked fields and possible errors to be checked by a student assistent


ttes automatisk\Analyseteam\kortlægningsscript_python\script_TFJ\Eform_Udvidet_Script\Boik_KFST.py
temp.py
input_til_script.py
Boik_KFST.py
opdelKontrakter.py
except AttributeError:
udbudDirektiv = ''
pass
#%%
""" Først og fremmest undersøges, hvor mange delkontrakter, der er.
Da den variabel, som Ordregiver kan bruge er dårlig, anvendes en anden tilgang,
hvor antallet af "Betegnelse" der indeleder en ny delkontrakt anvendes som indikator
Der anvendes samme tilgang som i Kontrakttildelingssektionen
Vi ønsker, at finde span med id='id1-II.' dens parent's parent er netop
Del II: Genstand
"""
""" VARIABEL Antal delkontrakter """
""" VARIABEL Opdelt udbud """
delIIGenstandSuppe = boikReqSuppe.find("span", {"id" : "id1-II."}).parent.parent
delIIGenstand = delIIGenstandSuppe.getText()
# der fratrækkes 1 da den første Betegnelse altid er der uanset antal delkontrakter
Antal_delkontrakter_kortlagt = delIIGenstand.count('Betegnelse:') - 1
if Antal_delkontrakter_kortlagt == 1:
opdelt_udbud_BOIK = 'nej'
# hvis der er flere delkontrakter end 1, da er det opdelt
elif Antal_delkontrakter_kortlagt > 1:
opdelt_udbud_BOIK = 'ja'
"""
Det sker sommetider, at "Betegnelse" ikke fremgår af afsnittet. Så rettes den dog til
"""
elif Antal_delkontrakter_kortlagt < 1:
Kommentar = tilføjKommentar(Kommentar, 'Usikkerhed om antallet af delkontrakter på udbuddet. ', 'Vigtigt')
Antal_delkontrakter_kortlagt = str(1)
Antal_delkontrakter_kortlagt = tilføjKommentar(Antal_delkontrakter_kortlagt, '', 'Vigtigt')
opdelt_udbud_BOIK = 'nej'
# Antal delkontrakter kortlagt konverteres til string for at arbejde med den senere
Antal_delkontrakter_kortlagt = str(Antal_delkontrakter_kortlagt)
"""
Her tjekkes om enten udbudsbekendtgørelse eller BOIK siger opdelt udbud
og i så fald antages det, at der er tale om opdelt udbud.
"""
if opdelt_udbud_BOIK == 'ja' or opdelt_udbud_Udbud == 'ja':
Opdelt_udbud = 'Ja'
else:
Opdelt_udbud = 'Nej'
# Laver dummy, der gør det muligt, at tilføje delkontraktadskiller. 1 når opdelt 0 ellers
if Opdelt_udbud == 'Ja':
DummyOpdelt = 1
else:
DummyOpdelt = 0
""" Rettelse af Antal_delkontrakter_kortlagt i tilfælde af at ordregiveren ikke har
udfyldt det igen på BOIK'en sørges for at, hvis antallet af kontrakttildelinger matcher antallet af
delkontrakter på udbuddet, da antages det at, der er ligeså mange delkontrakter på BOIK'en.
Der tilføjes dog også en kommentar.
"""
spanKontrakttildelinger = boikReqSuppe.findAll("span", {"id" : re.compile('id\d*-V\.')})
Source
Console
Object
Usage
Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.
Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in Preferences > Help.
New to Spyder? Read our tutorial
Help
Variable Explorer
Plots
Files
Console 1/A
Python 3.10.13 | packaged by Anaconda, Inc. | (main, Sep 11
13:15:57) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more informati
IPython 8.15.0 -- An enhanced Interactive Python.
In [1]:
IPython Console
History
conda (Python 3.10.13)
Completions: conda
LSP: Python
Line 41, Col 23
UTF-8
CRLF

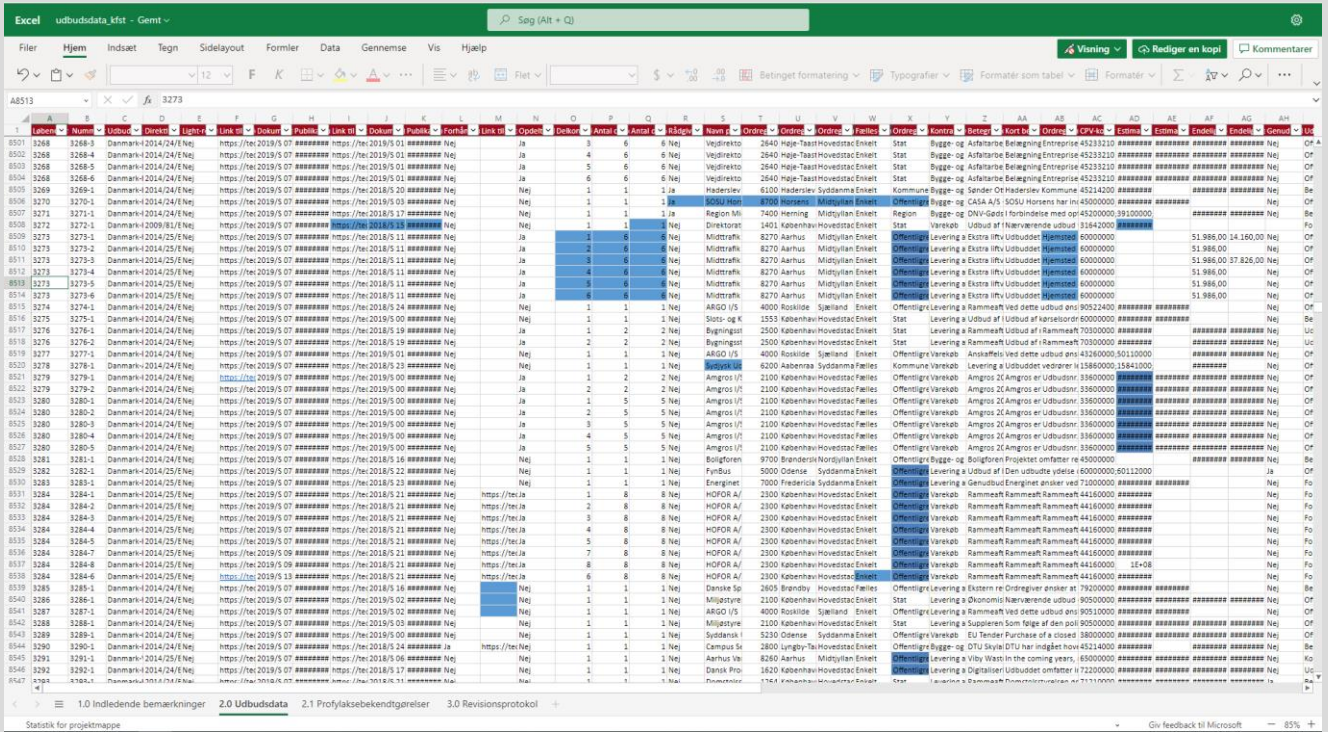
2019 – Automation of mapping data

Lessons learned

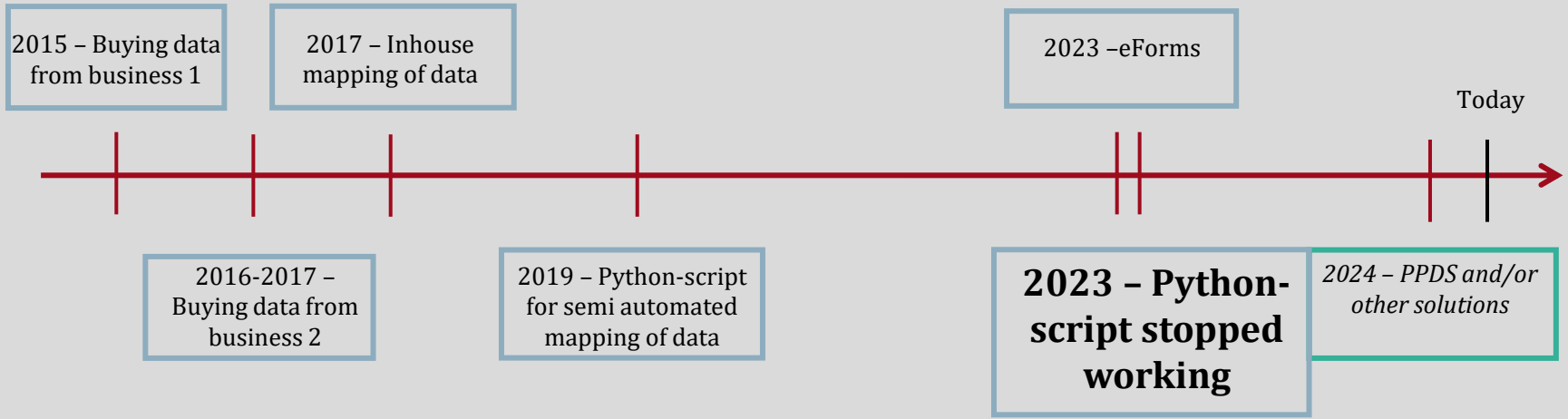
- Enormous amount of time saved
- The script optimized and streamlined mapping and quality assurance
- Additional manual quality assured an updated script and room for “weird tenders”
- Web-scraping is sensitive to change
- You need both an understanding of the filling in of notices and programming for an optimal set-up



Made available to the public in 2022



Timeline of procurement data



2023 – Script stopped working

Purpose and approach

- We want an automated data collection and quality assurance setup
- We are in the process of defining our new set-up

Lessons learned

- Data scientist are a limited ressource
- Ressources for attaining procurement data by programming must be prioritized

What have been the effects on quality?

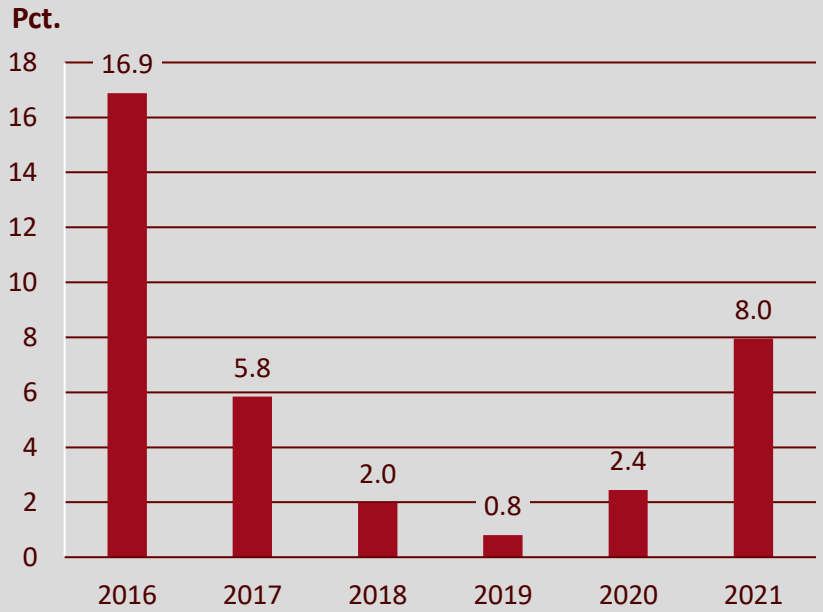
Effects of quality assurance

- Higher quality data
- Data has become usable for our needs
- Alterations and added information in every tender
- Measurable effects
  - Missing CAN's
  - Filled in values

Procedure for CAN-reminders

- Email to procuring authorities that have not published a CAN after 1,5 years
- High succes rate
- We found unlinked tenders and authorities published CAN's after contact
- The procedure stopped in 2021

Figure 1 - Overview of the share of missing CANs over time (2016-2021)



Fields that have been filled in

- All variables have been filled in to some extent. I have focused on a few key variables:

Total value from CAN	Winning bidder	Business registry number
• 9 percent of tenders	• 5 percent of tenders and lots	• 48 percent of tenders and lots

Generel effects of data mapping and quality assurance

High data quality that enables complex analyses and evaluations of our procurement law and practices

Semi-live data that can be used to inform the political level and provide legal experts with insights into practices

Awareness of data capabilities and limitations

A more complete overview of Danish tenders through increased number of CANs

Engagement and understanding amongst actual users of the notice-forms by contact

Danish Competition and Consumer Authority