



Python Ireland

# E-ink dashboards everywhere

Marcel-Jan Krijgsman



# Introduction

Marcel-Jan Krijgsman

Senior Data Engineer for DIKW Intelligence

Working with Python since 2017

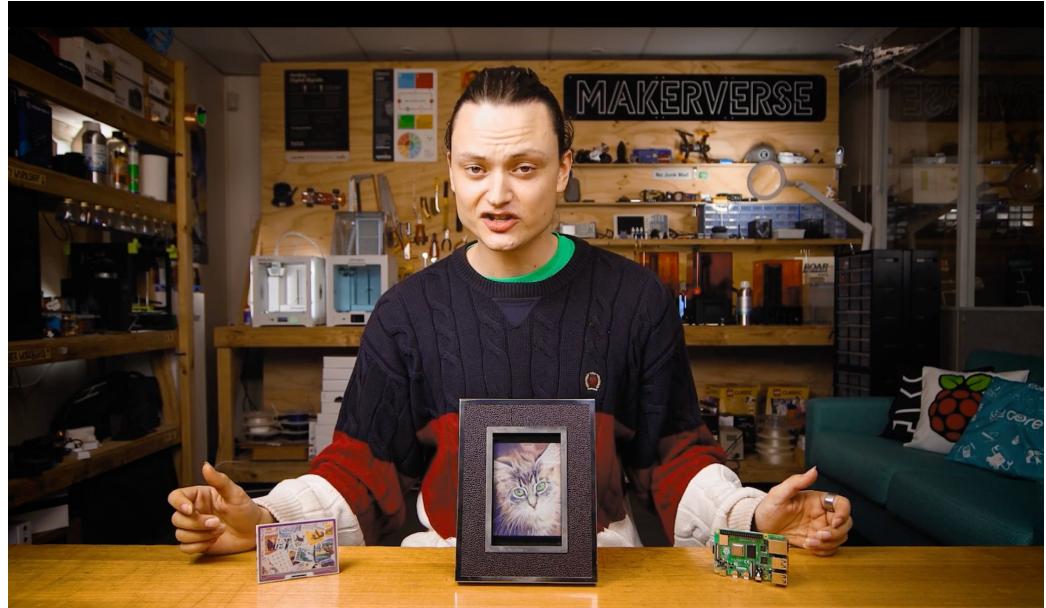
For work, but also for hobby.



# How it started..

Learned that e-ink displays existed via Hacker News.

Saw this video from Core Electronics



# What's this? The 1980s?

Available in black and white and 7-colors.

Resolutions are a bit low: 640x400 and 800x480.

Sounds like my old EGA card!

Refresh rates in .. multiple seconds.



# Actually e-ink displays are quite cool

Renders images quite well.

When not refreshing they use no energy.

Runs on a Raspberry Pi.



# Uses for e-ink displays

## Applications:

- You want some data or image to show.
- You want it to look nice, but it doesn't have to be of high photo quality.
- You don't need up to second updates. > minute intervals are fine.



# E-ink brands



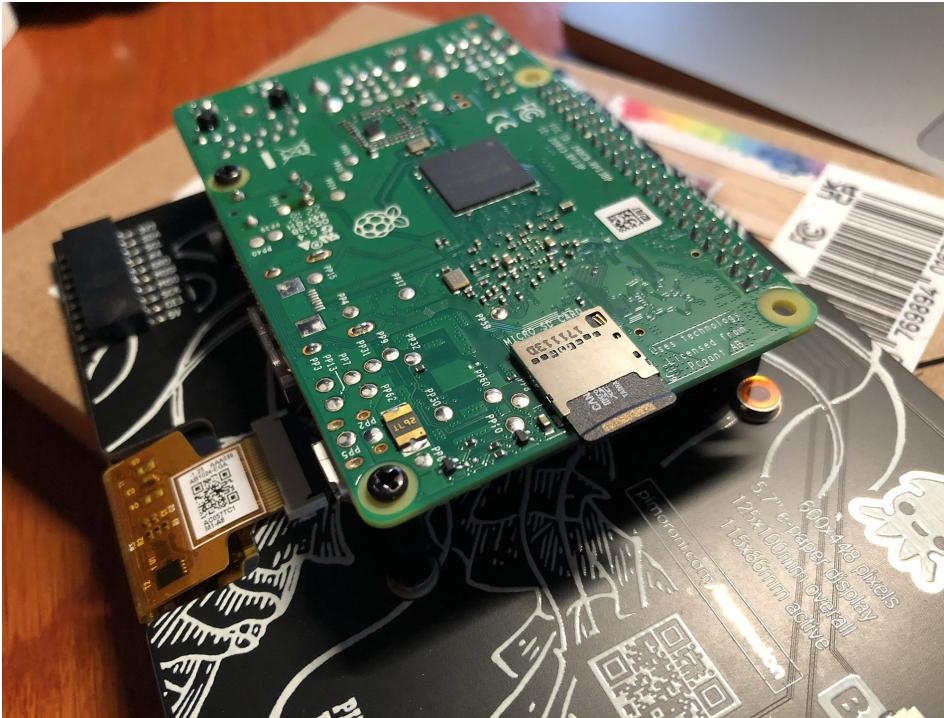
Inky Impression 7-color eHAT



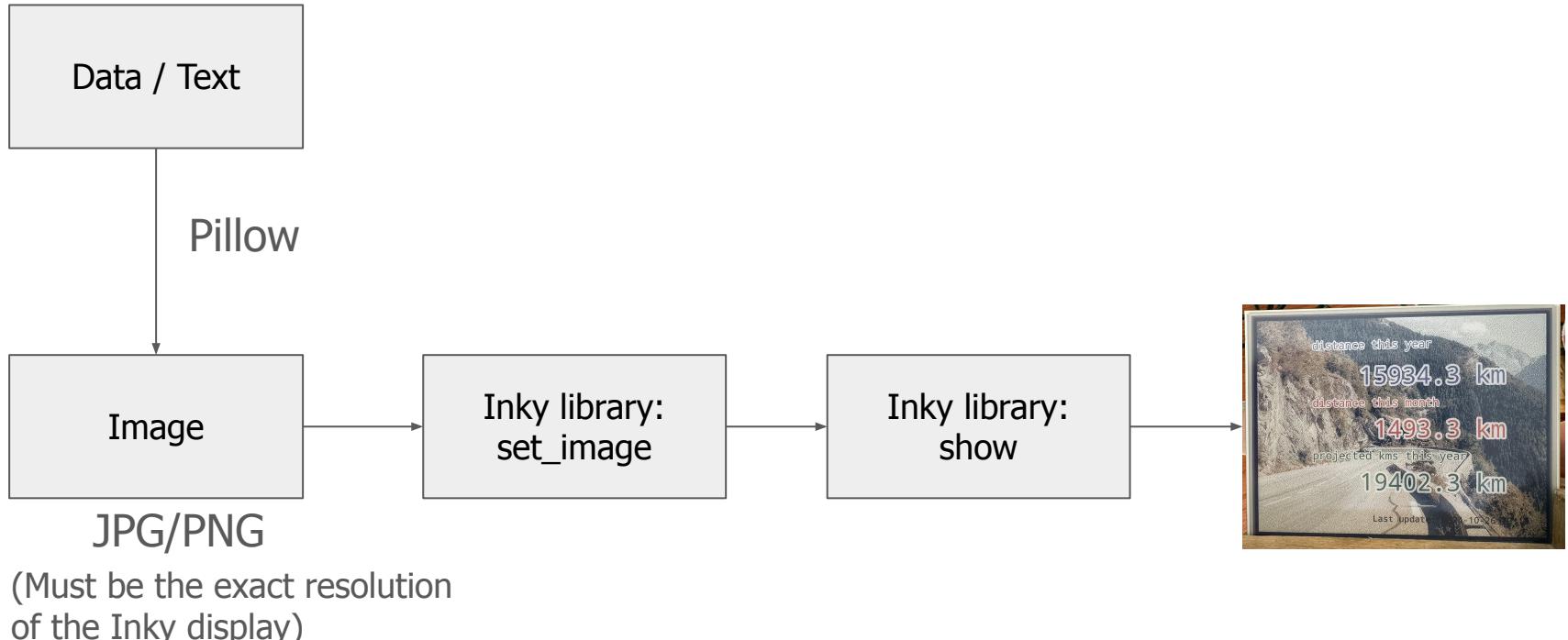
E Ink Spectra 6 (E6) Full color  
E-Paper Display

# No soldering: eHATs

Just click your e-ink display on the GPIO port of a Raspberry Pi.



# How to get an image on an Inky display



# Inky in Python

```
pip install inky[rpi]
```

## Code:

```
from inky.auto import auto

INKY_DISPLAY = auto()
image_name = "myimage.jpg"
INKY_DISPLAY.set_image(image_name)
INKY_DISPLAY.show()
```

# The possibilities are endless ...

And yet..



Photo frame



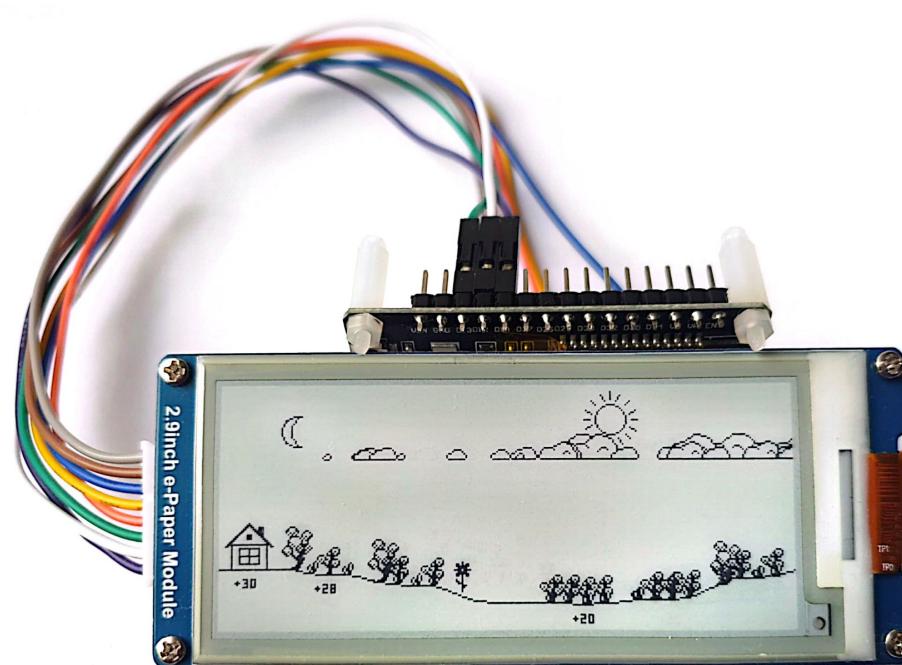
Calendar



Weather station

# Granted, some weather station apps are cool

[https://github.com/lds133/weather\\_landscape](https://github.com/lds133/weather_landscape)



# With the power of Python

---



You can show any kind of image and data.

So let's put our thinking caps on.





# E-ink project 1: A cycling dashboard



# Strava has my cycling data. Can I have it?



Yes! Strava has a developer API:

<https://developers.strava.com/>

Oh but I need to authenticate...

Luckily someone already figured that out, and  
blogged about it. His name is Graziano Fuccio.



Welcome guys to a new episode of the #getyourdata series with Python. Here we'll learn how to interact with Strava API through RESTful requests to get your sports data. This could be an interesting chance to improve your Python skill and your ability to scrape and collect data.

## The Swagger Client

In Strava's API [Getting Start Guide](#), developers suggest to use [Swagger Playground](#) to interact with Strava API submitting HTTP requests and processing the response. I tried, but I don't like the approach. Too many complications to installing the Swagger Client that maybe we will use only this time in our entire developer's life. Instead, we can proceed with an easy and elegant Python solution!

## Before Starting

Before starting, you need to register an account on [Strava](#). Don't worry, it is not required a premium account to use API.



# Once authenticated..

It's a matter of looking through the available data.

And before you know it:



# Give it a little make up.

Wooden frame by my father.

Christmas themed cycling background by  
Stable Diffusion.

Code available:

<https://github.com/Marcel-Jan/Stravalinky>



# Icing on the cake...

Voted 90% as “Hack” in Hack vs. Bodge on the Global Cycling Network



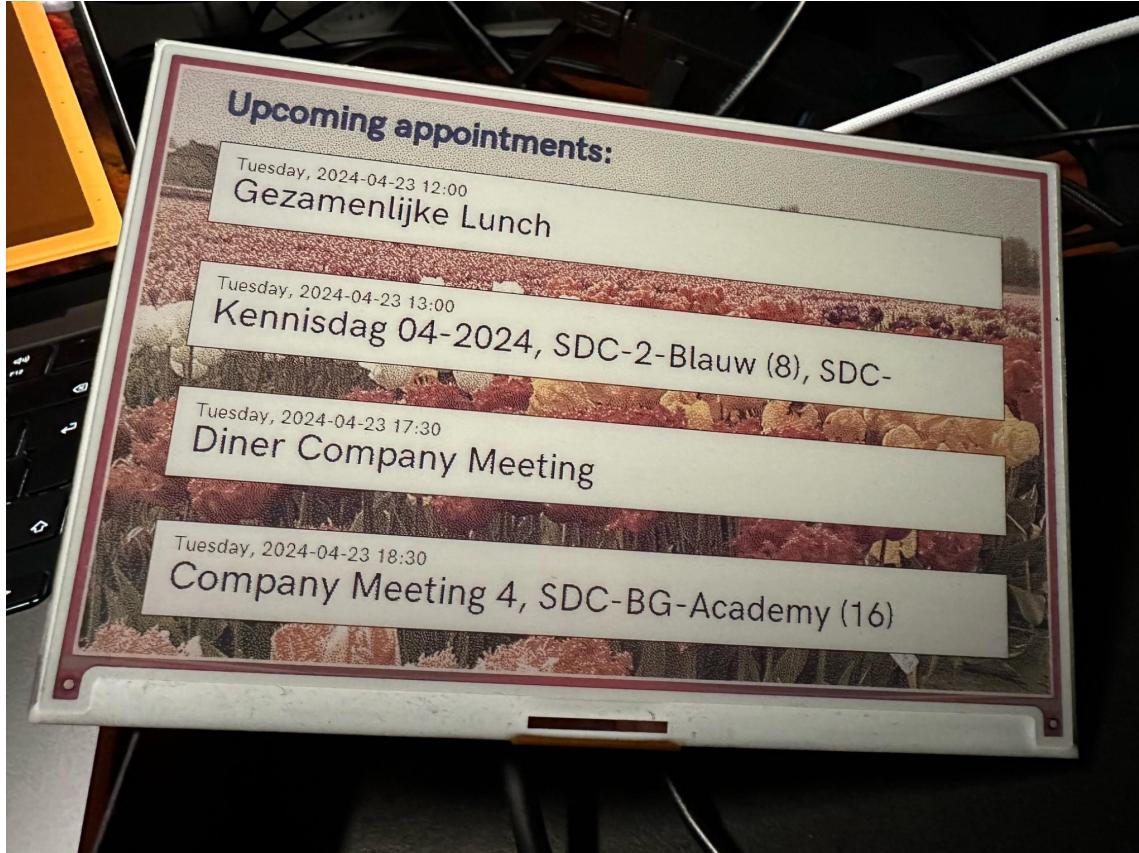


# E-ink project 2: A wall calendar

(Yes, it falls in the 3 categories)



# My upcoming appointments





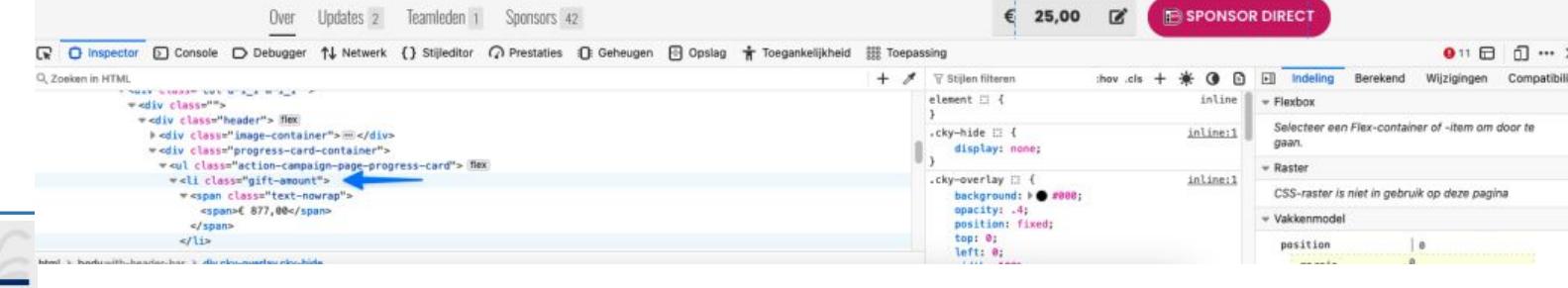
# E-ink project 3: A donation tracker



# Getting donations for cancer research



# Scraping. Easy right?



The screenshot shows the browser developer tools with the 'Inspector' tab selected. The DOM tree on the left shows the HTML structure of the crowdfunding page, with a blue highlight on the 'li' element containing the amount '€ 877,00'. The right panel shows the CSS inspector with the 'Stijlen filteren' (Style filter) set to ':not(.cls)'. It lists several CSS rules, including '.element' (display: inline), '.cky-hide' (display: none), and '.cky-overlay' (background: #000; opacity: .4; position: fixed; top: 0; left: 0;). The 'Indeling' (Layout) tab is also visible in the top right.

# Scraping dynamic HTML

Used requests-html package.

How it works: it basically runs a Chromium browser in the background.

This works on my laptop. Success!



# On to the Raspberry Pi!

Now copy the Python code to the Raspberry Pi, install the packages and...

```
[INFO] Starting Chromium download.  
100% |████████████████████████████████| 183M/183M [00:17<00:00, 10.4Mb/s]  
[INFO] Beginning extraction  
[INFO] Chromium extracted to: /home/inky3/.local/share/puppeteer/local-chromium/1181205  
Traceback (most recent call last):  
  File "/home/inky3/PythonProjects/pidonationtracker/src/pidonationtracker/pidonationtracker.py",  
line 59, in <module>  
    donation_amount = get_donation_amount()  
[...]  
  
  File "/usr/lib/python3.9/subprocess.py", line 1823, in execute_child  
    raise child exception type(errno_num, err_msg, err_filename)  
OSErr: [Errno 8] Exec format error:  
'/home/inky3/.local/share/puppeteer/local-chromium/1181205/chrome-linux/chrome'
```



# The problem

requests-html runs a package called puppeteer.

puppeteer starts a Chromium browser.

But it's a version for Intel processors. Raspberry Pi's have ARM processors.

An ARM version of Chromium hasn't been released since 2018. Including the codecs the code crashed on.

The only way I could get this working: downgrade my Chromium back to 2018.



# Success..



# How do I get power and WiFi in the wild?



Power: powerbank!

WiFi: works via my phone's hotspot.



# Field test





# E-ink project 4: A star map



# An up to date star map every night



# Can I have a star map package please?



Turns out that star map/plot is also a kind of data visualization.

After some creative searching I found Starplot  
(<https://starplot.dev>)

It has a list of dependencies. One of them, cartopy, wouldn't load on the Raspberry Pi Zero 2.

A screenshot of a Google search results page for the query "python package starmap". The results are displayed in a dark-themed interface.

- Python Docs**  
<https://docs.python.org/library/multiprocessing.html#multiprocessing.Pool.starmap>  
multiprocessing is a package that supports spawning processes using an API similar to the threading module. The multiprocessing package offers both local and ...
- GeeksForGeeks**  
<https://www.geeksforgeeks.org/python-itertools-starmap/>  
Python - Itertools.starmap()  
13 Apr 2022 — The starmap() considers each element of the Iterable within another Iterable as a separate item. It is similar to map(). This function comes under the category ...
- PyPI**  
<https://pypi.org/project/starmap/>  
starmap  
30 Jun 2020 — pip install starmap. Copy PIP instructions. Latest version: Released: Jun 30, 2020. StarMap placeholder. Navigation.
- Stack Overflow**  
<https://stackoverflow.com/questions/how-to-get-multiprocessing-pool-starmap-to-return-iterable>  
How To get multiprocessing.Pool().starmap() to return iterable  
starmap returns the result in the order they were submitted, so after you get the results you can run the same loop but this time without needing to call func:  
1 answer · Top answer · Your starmap version and normal version are not equivalent. When us...  
How to use multiprocessing.Pool.map with multiple arguments · 26 Mar 2011  
python - Multiprocessing starmap creates duplicate objects · 23 Nov 2022  
Is my understanding of multiprocessing's starmap correct? · 19 Apr 2022  
Does python.multiprocessing have starmap? - Stack Overflow · 23 Jan 2019  
More results from stackoverflow.com
- Super Fast Python**  
<https://superfastpython.com/tutorials/multiprocessing-pool-starmap-in-python/>  
Multiprocessing Pool.starmap() in Python  
11 Jul 2022 — You can map a function that takes multiple arguments to tasks in the process pool via the Pool.starmap() method.  
How to Use Pool.starmap() · Example of Pool.starmap() · Example of Pool.starmap...
- GitHub**  
<https://github.com/weallem/STARmap>  
Software for processing and analyzing STARmap ...  
Software for processing and analyzing STARmap experiments. - weallem/STARmap ...



# Loading Cartopy



Everytime I tried pip install cartopy (poetry add cartopy) the load on the Raspberry Pi Zero 2 went up really high.

And then it became unresponsive. I was no longer able to ssh into it.

Turns out, it was swapping like crazy.

Solution: enlarge swap space (<https://pimylifeup.com/raspberry-pi-swap-file/>)

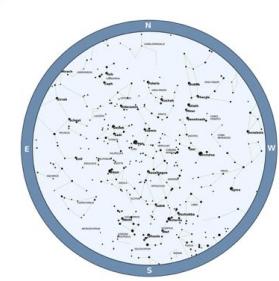
And all was good again.



# Starplot

Examples

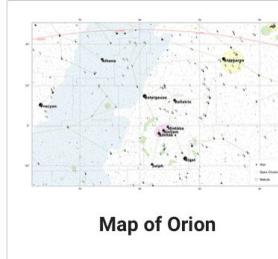
Starplot  
Overview  
Installation  
Examples  
Tutorial  
Reference  
Data Sources  
License  
Changelog  
Coming Soon  
About



Basic Star Chart



Detailed Star Chart

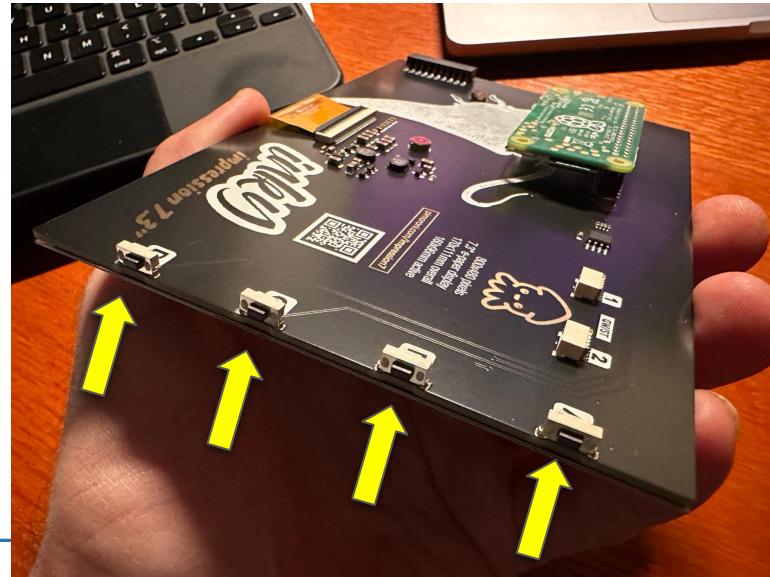


# Future development

Refine the star map so it shows the sky in one direction (north, west, east, south).

Did you know these e-ink displays usually have (4) buttons?

How convenient!





# Other ideas for e-ink displays



# Other applications I can think of



That tired old question: "Is this room booked?" ("And by whom?")

A display that shows if my train runs on time.

"How is traffic between home and my office?"





# E-ink displays: the future



# What I want in the future

Poster sized e-ink displays!

Until that time:





# Thank you!

