

**KOMPLEXNÍ
VIZUALIZACE
DAT A PŘEHLED
PROGRAMOVACÍCH
JAZYKŮ PRO
VIZUALIZACI**

#01 RESTY

A GOOGLE DOCS

B GOOGLE CHARTS

#02 CO JE KOMPLEXNÍ VIZUALIZACE?

**Zeptáme se
Googlu**

**„Předmětem veřejné
zakázky bude navrhnout
komplexní vizualizaci města
Šumperka, jejímž základem
bude značka a logotyp pro
město Šumperk, případně
pro organizace jím založené
a zřízené.“**

— Mgr. Zdeněk Brož

“Functional visualizations are more than innovative statistical analyses and computational algorithms. They must make sense to the user and require a visual language system that uses colour, shape, line, hierarchy and composition to communicate clearly and appropriately, much like the alphabetic and character-based languages used worldwide between humans.”

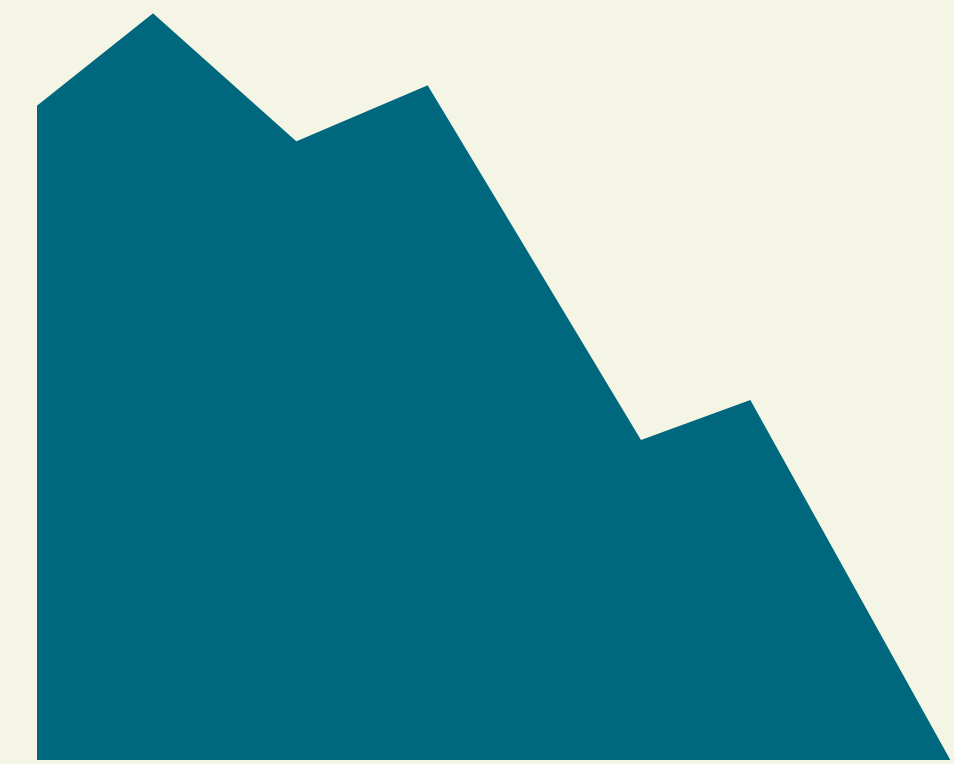
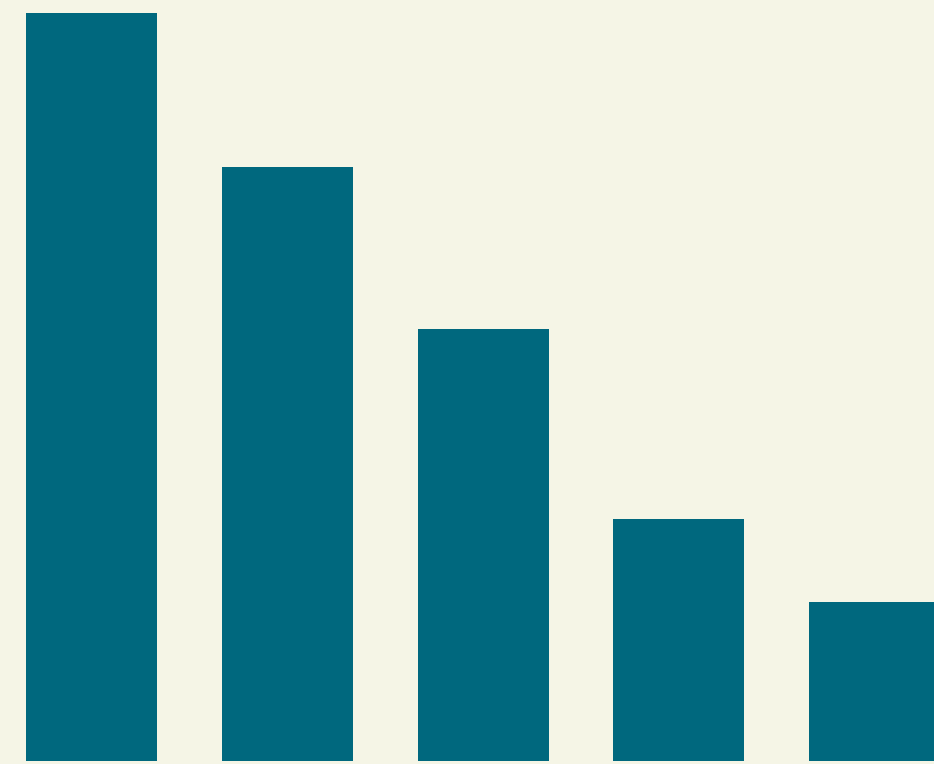
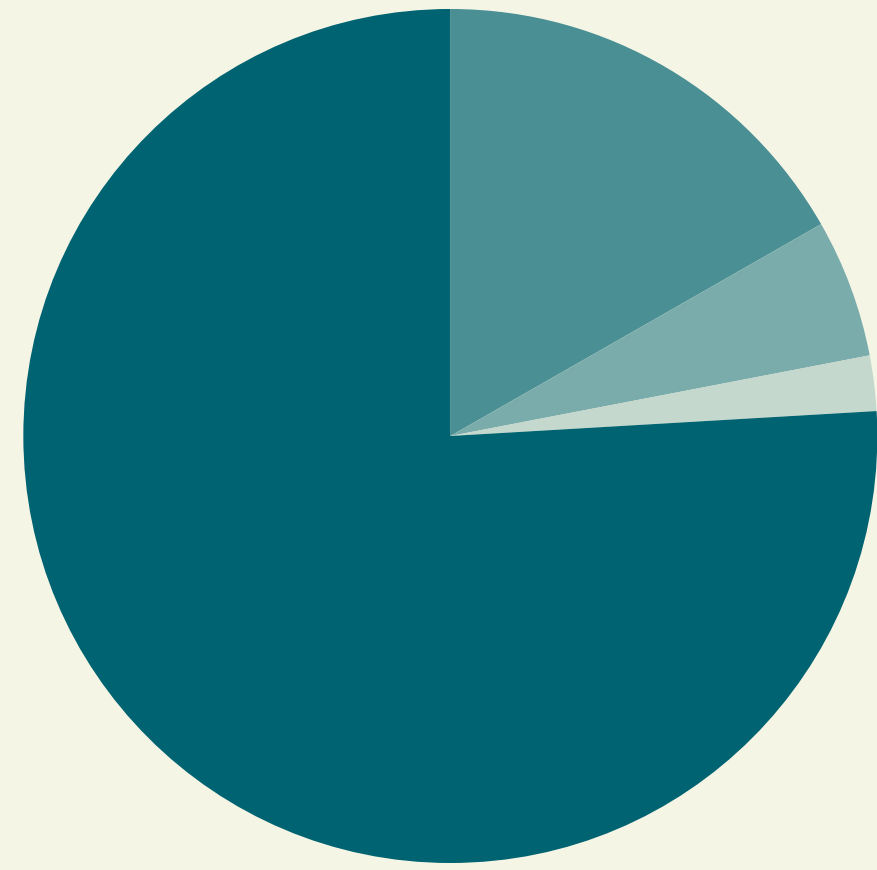
— Matt Woolman, *Digital Information Graphics*

a **Interaktivní
vizualizace
relativně
jednoduchých
dat**

**b Statická
vizualizace
komplexních
dat**

c Interaktivní vizualizace komplexních dat

Takže tohle to není

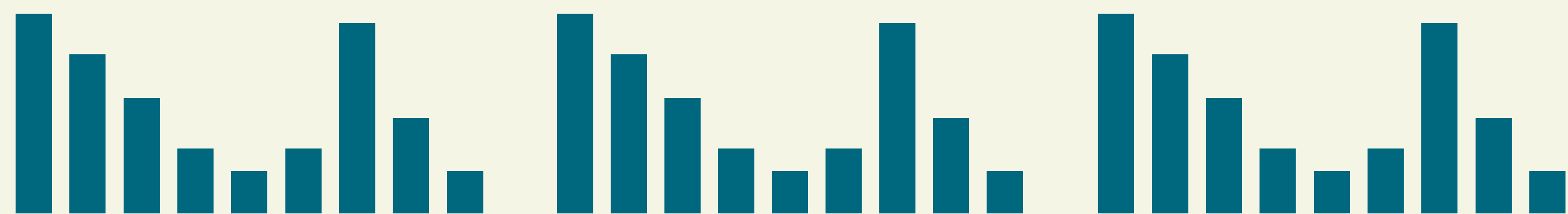
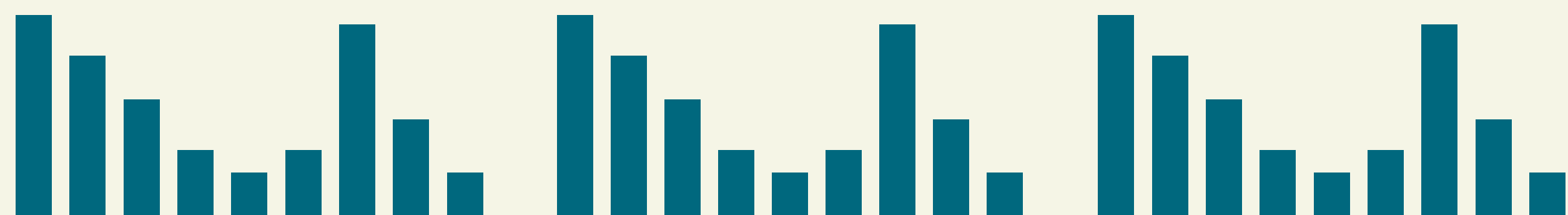


Ale tohle už by třeba být mohlo

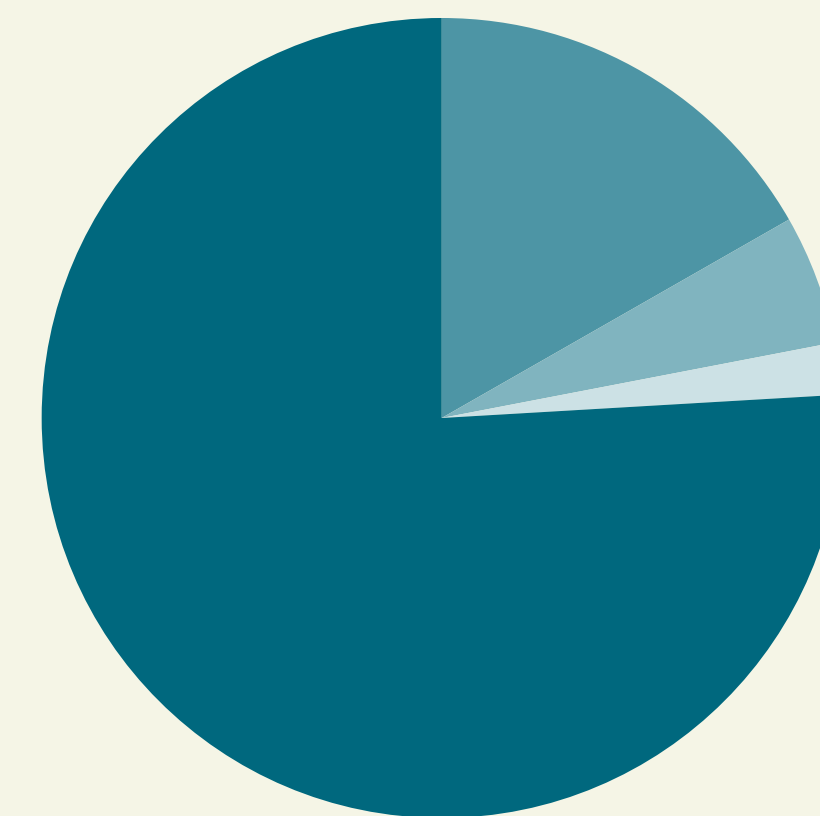
1. 2014 2. 2015 3. 2016 4. 2017 5. 2018 6. 2019 7. 2020 8. 2021 9. 2022 10. 2023



1. 2014 2. 2015 3. 2016 4. 2017 5. 2018 6. 2019 7. 2020 8. 2021 9. 2022 10. 2023



1. 2014 2. 2015 3. 2016 4. 2017 5. 2018 6. 2019 7. 2020 8. 2021 9. 2022 10. 2023



UK Temperature History

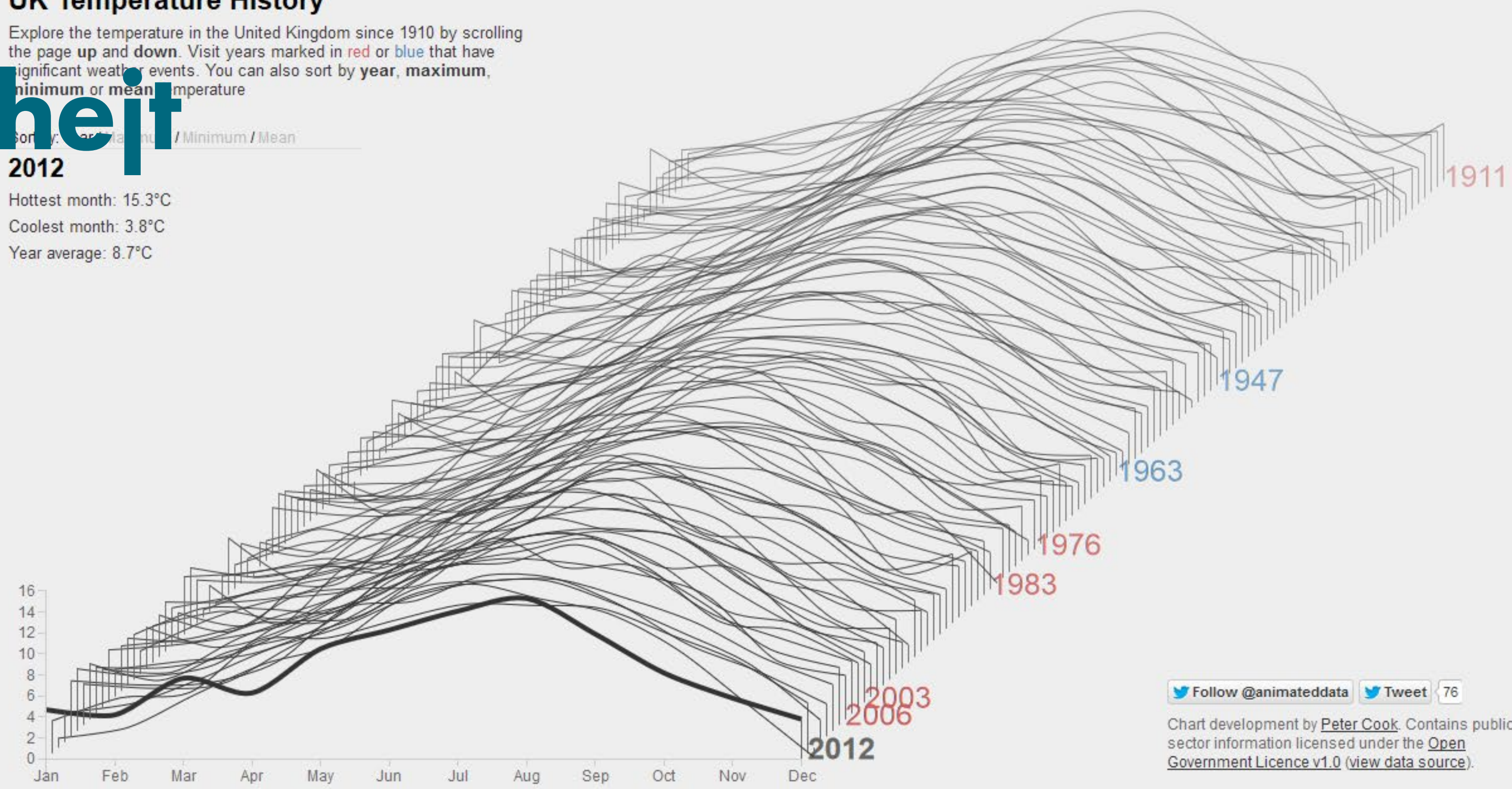
Explore the temperature in the United Kingdom since 1910 by scrolling the page up and down. Visit years marked in red or blue that have significant weather events. You can also sort by year, maximum, minimum or mean temperature

minihejt

Sort by: Year / Maximum / Minimum / Mean

2012

Hottest month: 15.3°C
Coolest month: 3.8°C
Year average: 8.7°C



Follow @animateddata Tweet 76

Chart development by Peter Cook. Contains public sector information licensed under the Open Government Licence v1.0 (view data source).



Brno

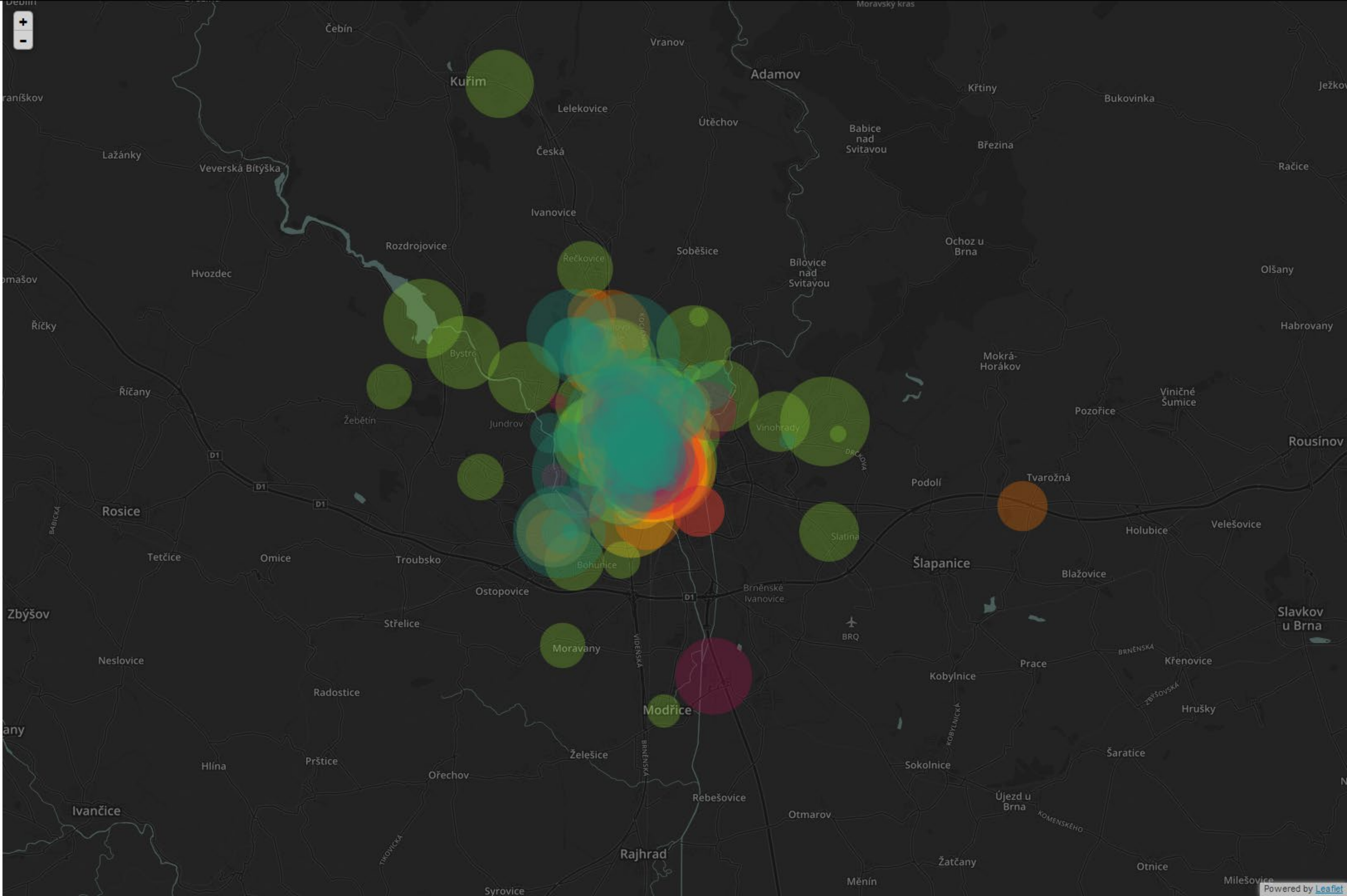
VISUALIZE!

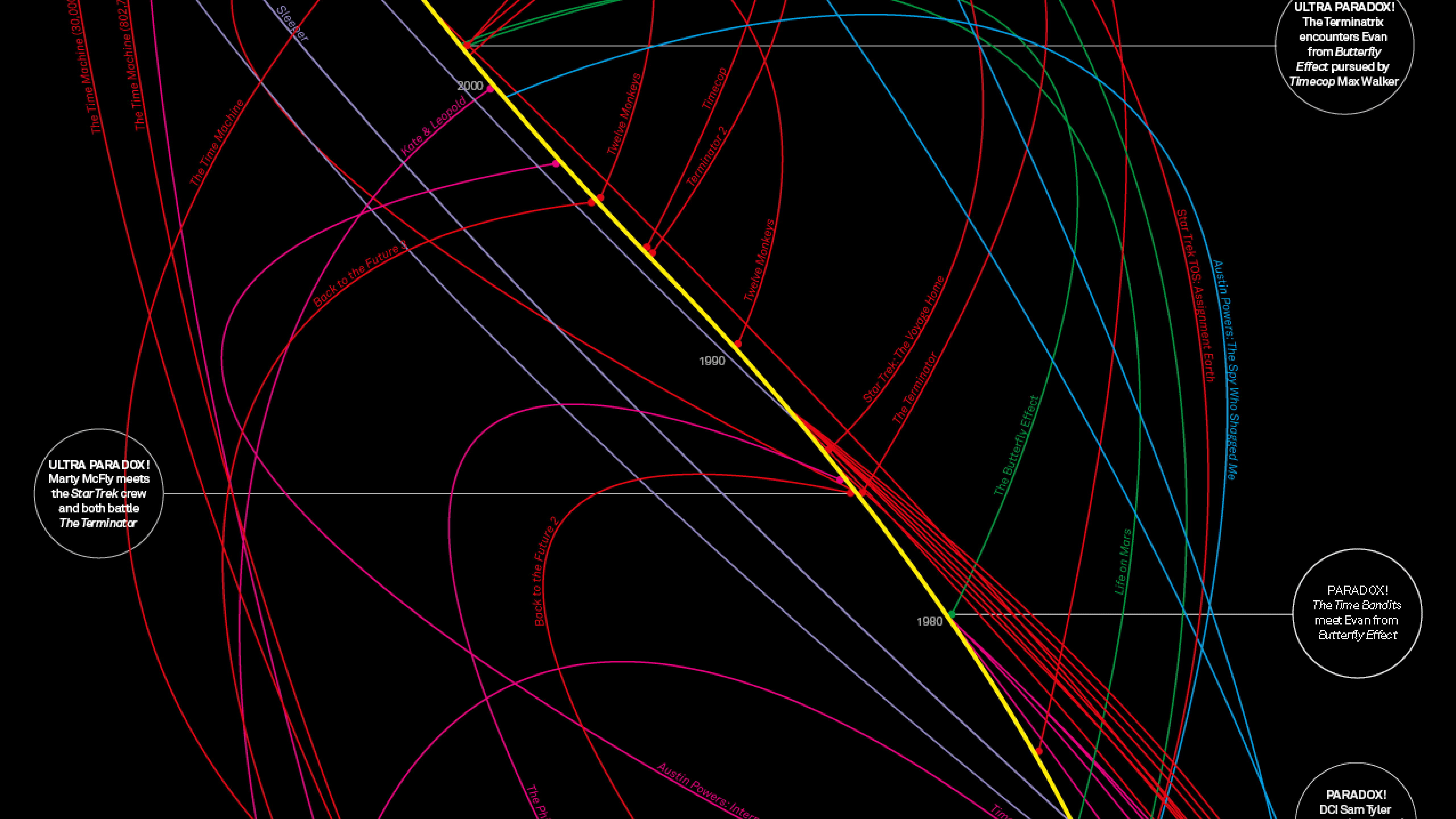
Have you ever wondered how would a city look, if you visualized it's venues by their usage and popularity? Kind of a urbanist study so to speak. Well, that's exactly what this visualization does... not show.

What it does show is all-but-logical selection of venues from five categories (50 venues each, Foursquare limitation, don't ask) scraped from Foursquare API. The venues are then placed into a map with a scale corresponding to their "popularity" (homebrewed equation: popular are those places, which have fairly large number of visitors, who tend to return there).

LEGEND

- Outdoors & Recreation
- Food
- Arts & Entertainment
- College & University
- Nightlife





ULTRA PARADOX!
 The Terminatrix
 encounters Evan
 from Butterfly
 Effect pursued by
 Timecop Max Walker

ULTRA PARADOX!
 Marty McFly meets
 the Star Trek crew
 and both battle
 The Terminator

PARADOX!
 The Time Bandits
 meet Evan from
 Butterfly Effect

PARADOX!
 DCI Sam Tyler

2000

1990

1980

The Time Machine

Sleeper

Kate & Leopold

Twelve Monkeys

Terminator 2

Timecop

Twelve Monkeys

Star Trek: The Voyage Home
 The Terminator

The Butterfly Effect

Life on Mars

Star Trek TOS: Assignment Earth

Austin Powers: The Spy Who Shagged Me

Back to the Future 3

Back to the Future 2

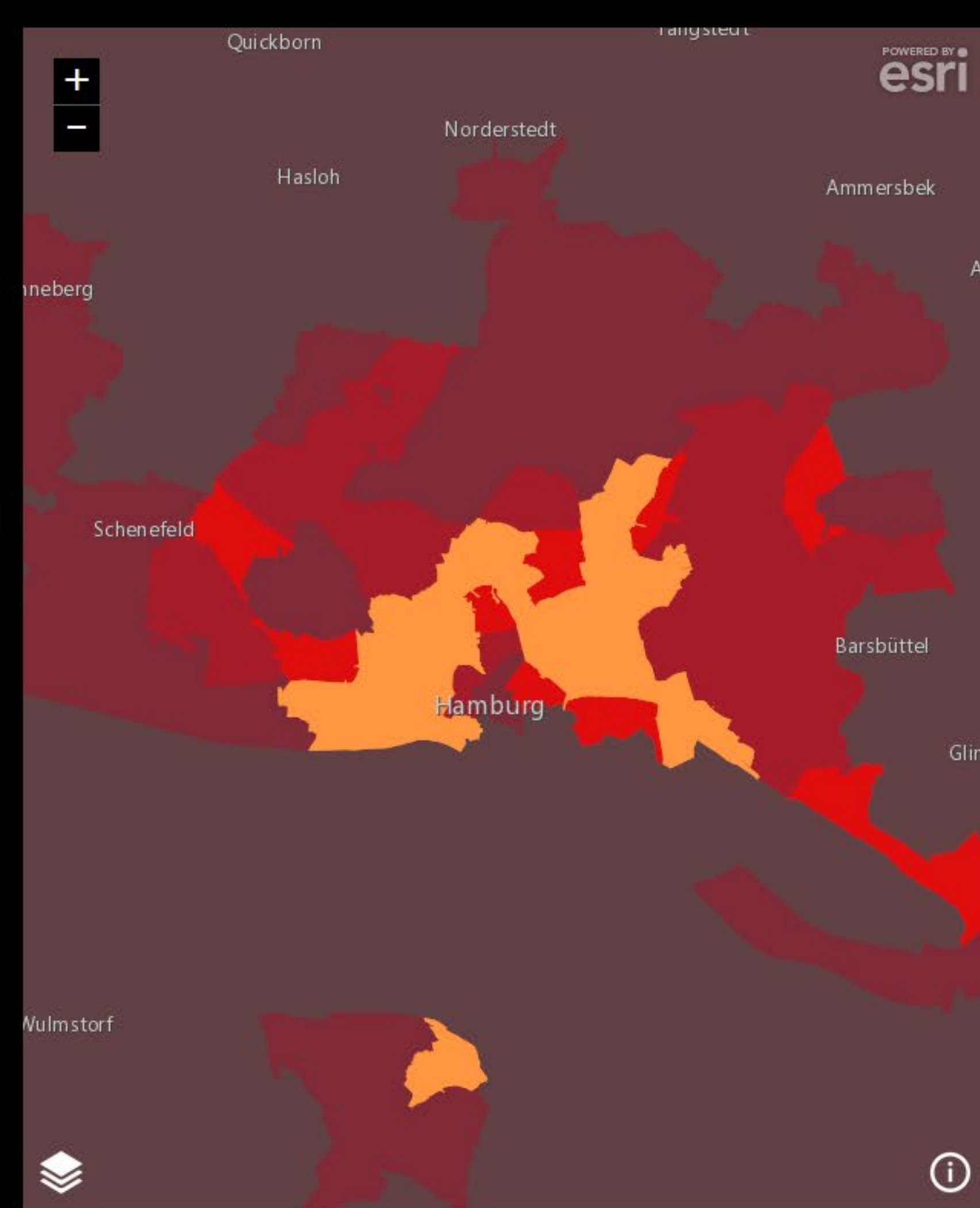
The Time Machine (30,000)

The Time Machine (802.7)

The Phil

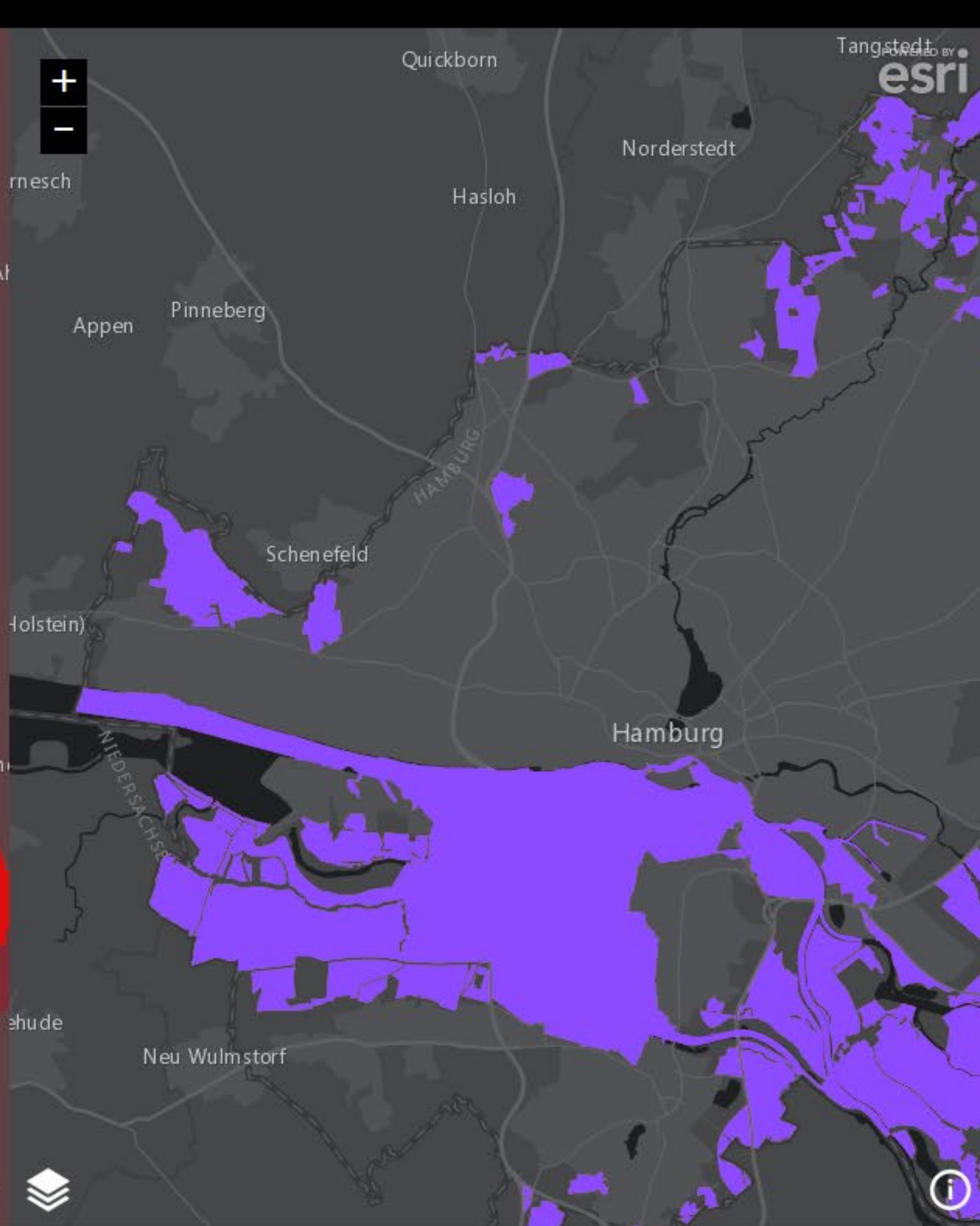
Austin Powers: Intern

Time



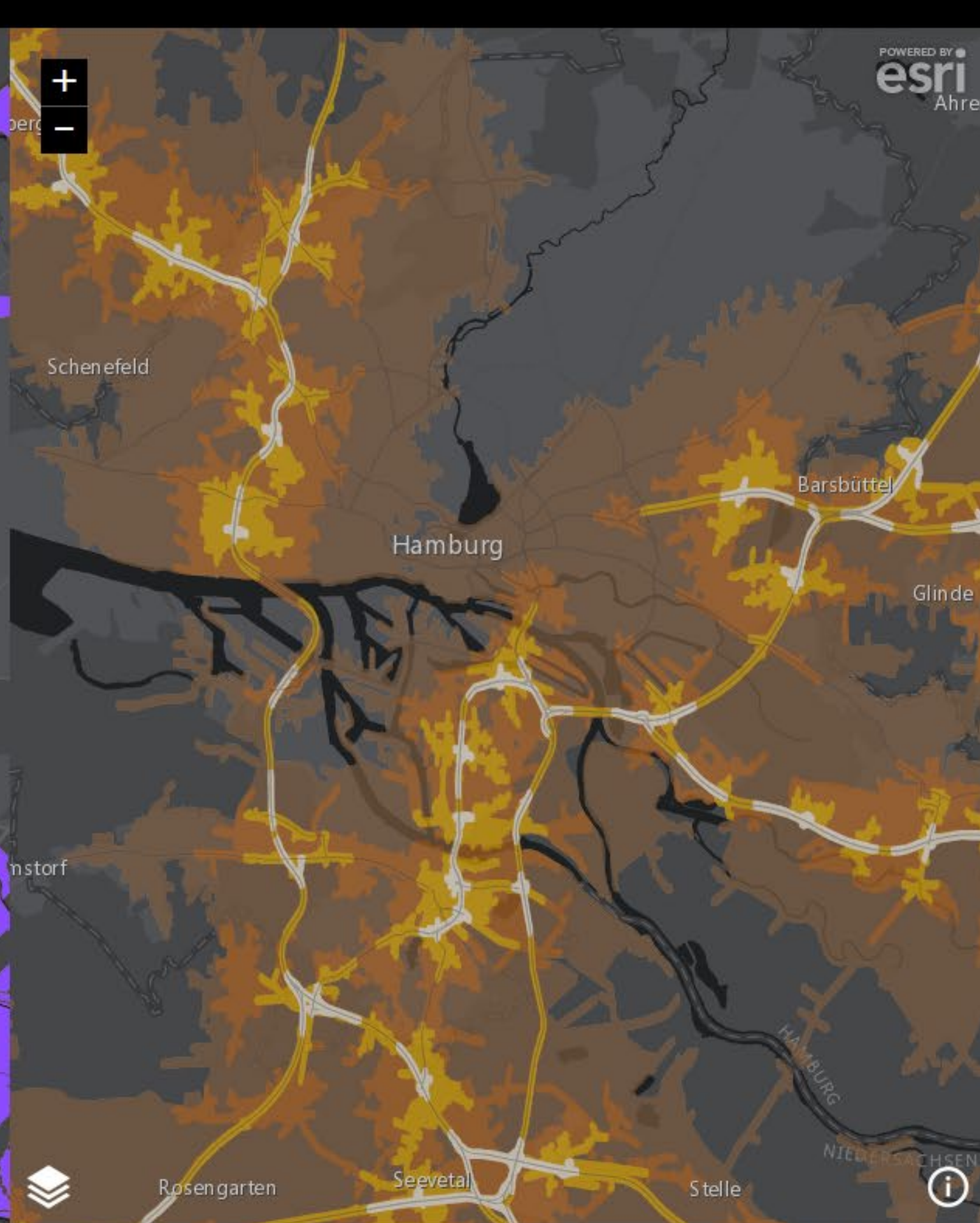
People /
Population Density ^

Touch to change



Work /
Industrial ^

Touch to change



Movement /
Highway Access ^

Touch to change



Selected City /
Hamburg



Abu Dhabi



Accra



Algiers



Atlanta



Auckland



Bangkok



Beijing

[Browse All Cities](#)

[Explore 3 Cities](#)

**No a jak něco
takového
udělat?**

#03 ÚVOD DO PROGRAMOVÁNÍ

A ZÁKLADNÍ POJMY

Proměnná

zastupuje nějakou hodnotu, která se při běhu programu může měnit.

Pole

Datová struktura. Obsahuje několik prvků, které lze adresovat (vybrat).

Funkce

Známe z matematiky, pro nějaký vstup z definičního oboru vrátí výsledek.

Metoda

Může mít vstup, ale nic nám nevrátí, jen něco změní ve vnitřním stavu programu.

Podmínka

If ... then ... else ...

Základní konstrukce. Pokud něco platí, udělej tohle, jinak udělej tohle.

Cyklus

For ... do ...

While ... do ...

Druhá základní konstrukce. Provádí danou akci, dokud platí nějaká podmínka.

Metoda

Může mít vstup, ale nic nám nevrátí, jen něco změní ve vnitřním stavu programu.

Podmínka

If ... then ... else ...

Základní konstrukce. Pokud něco platí, udělej tohle, jinak udělej tohle.

Cyklus

For ... do ...

While ... do ...

Druhá základní konstrukce. Provádí danou akci, dokud platí nějaká podmínka.

**K tomu ještě
dokumentace a víc
není potřeba. Stačí
jen vhodně skládat
k sobě.**

B FORMÁTY

DAT

CSV

Comma separated values — známe, hodnoty oddělené čárkou, v prvním řádku popisky.

JSON

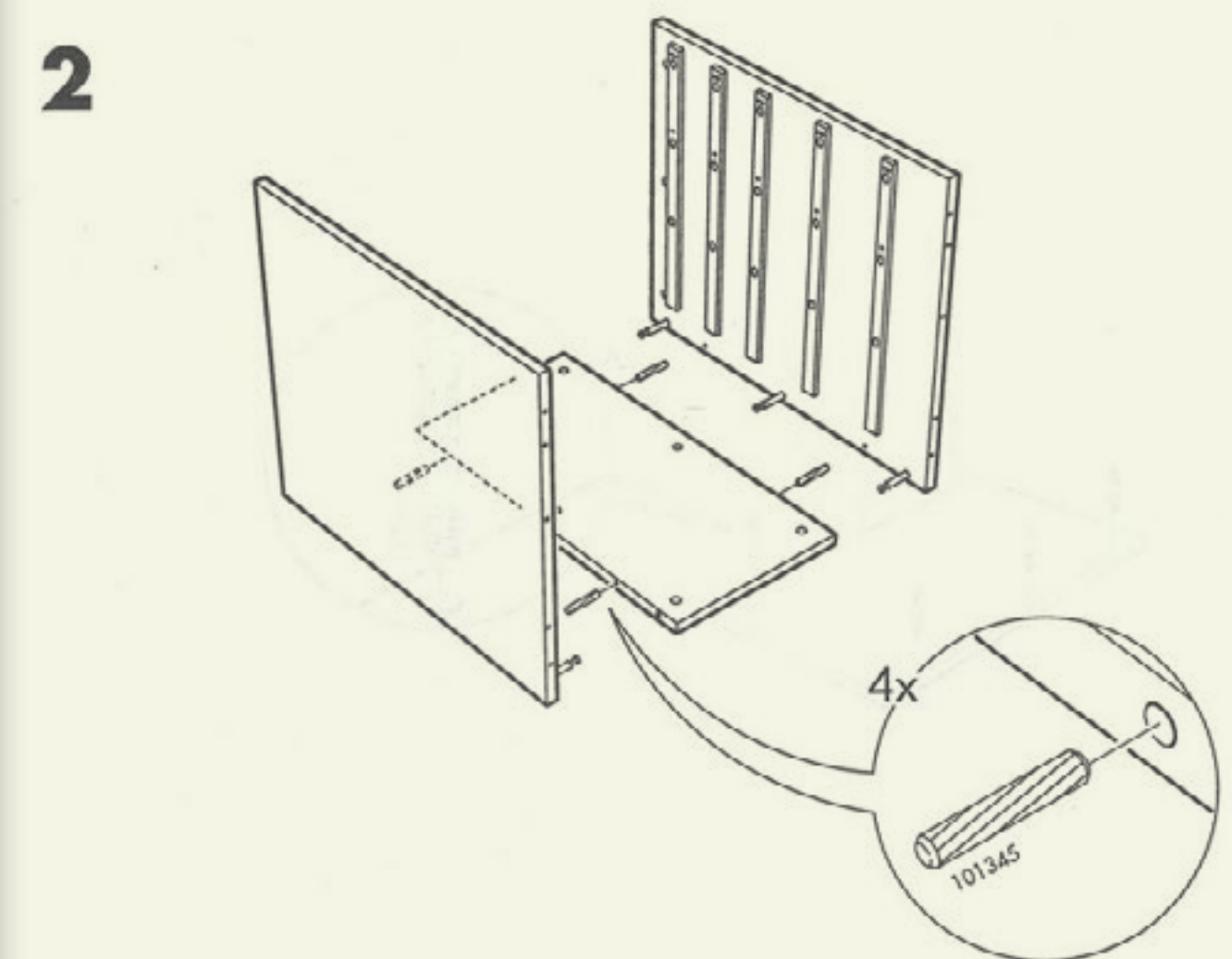
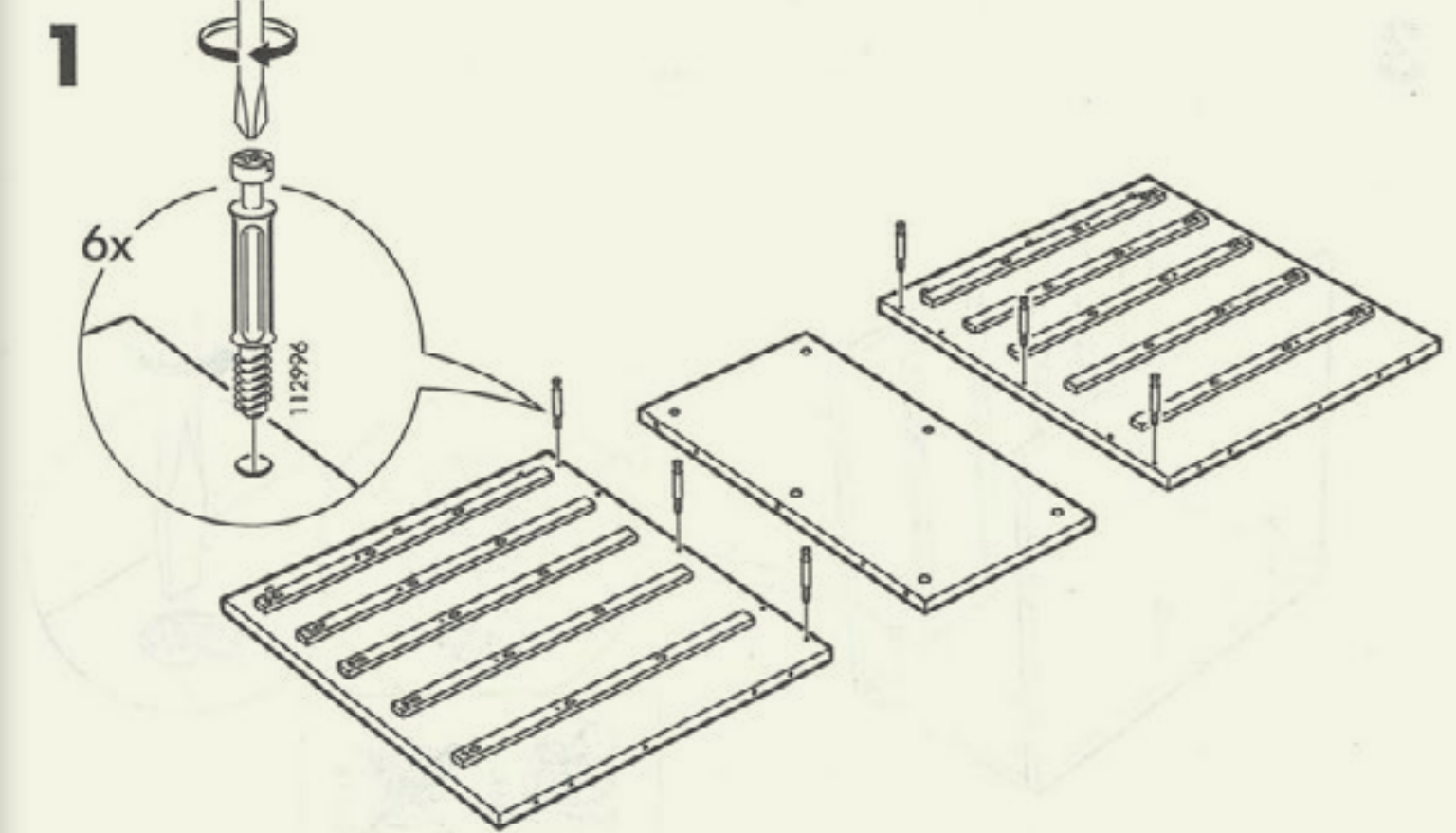
JavaScript Object Notation — strukturou připomíná objekt, kterému jsou připsané různé vlastnosti.

XML

Podobné jako JSON, má odlišnou syntax.

C PŘEHLED PROGRAMOVACÍCH JAZYKŮ PRO VIZUALIZACI DAT

PROGRAMOVÁNÍ Z NIČEHO

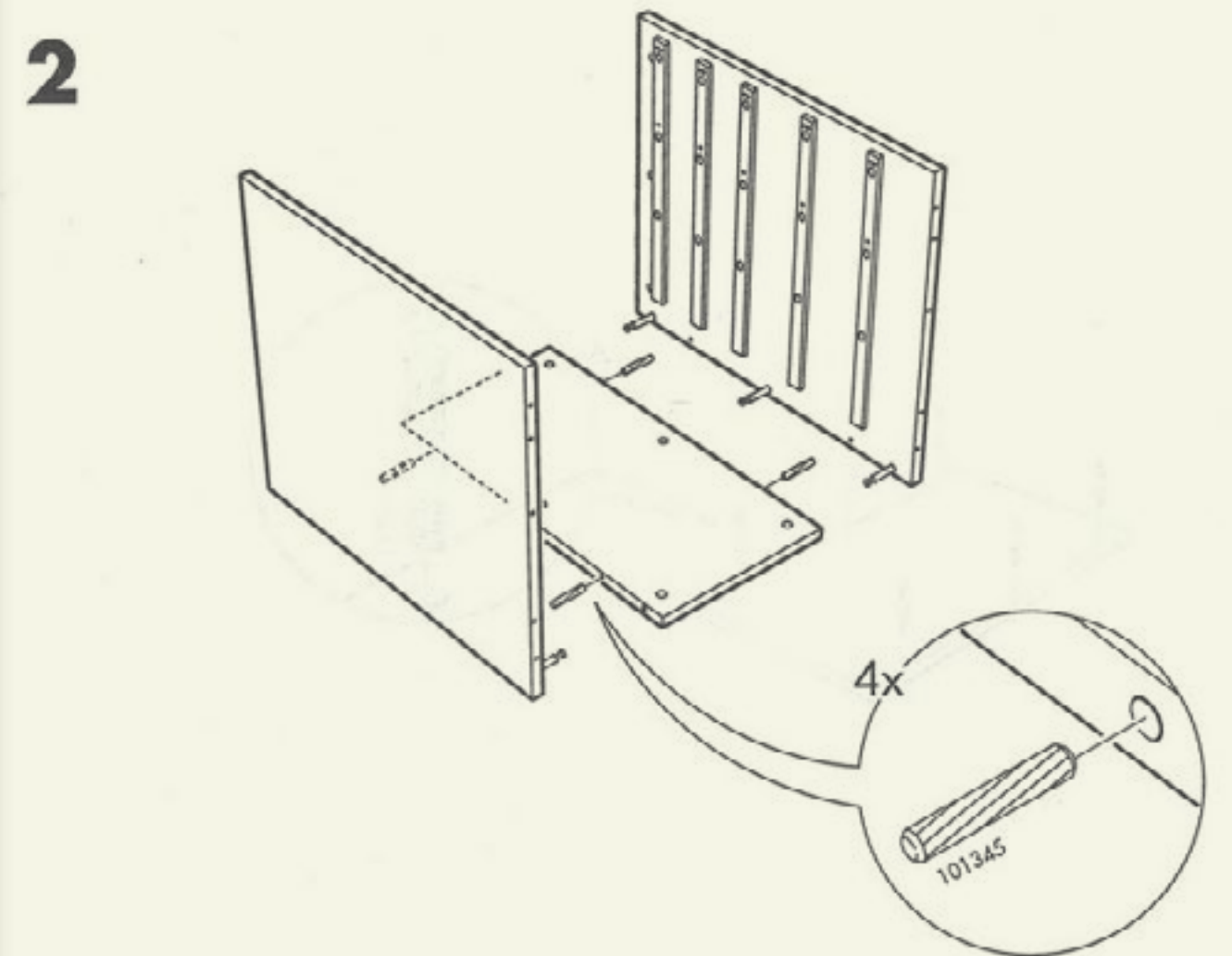
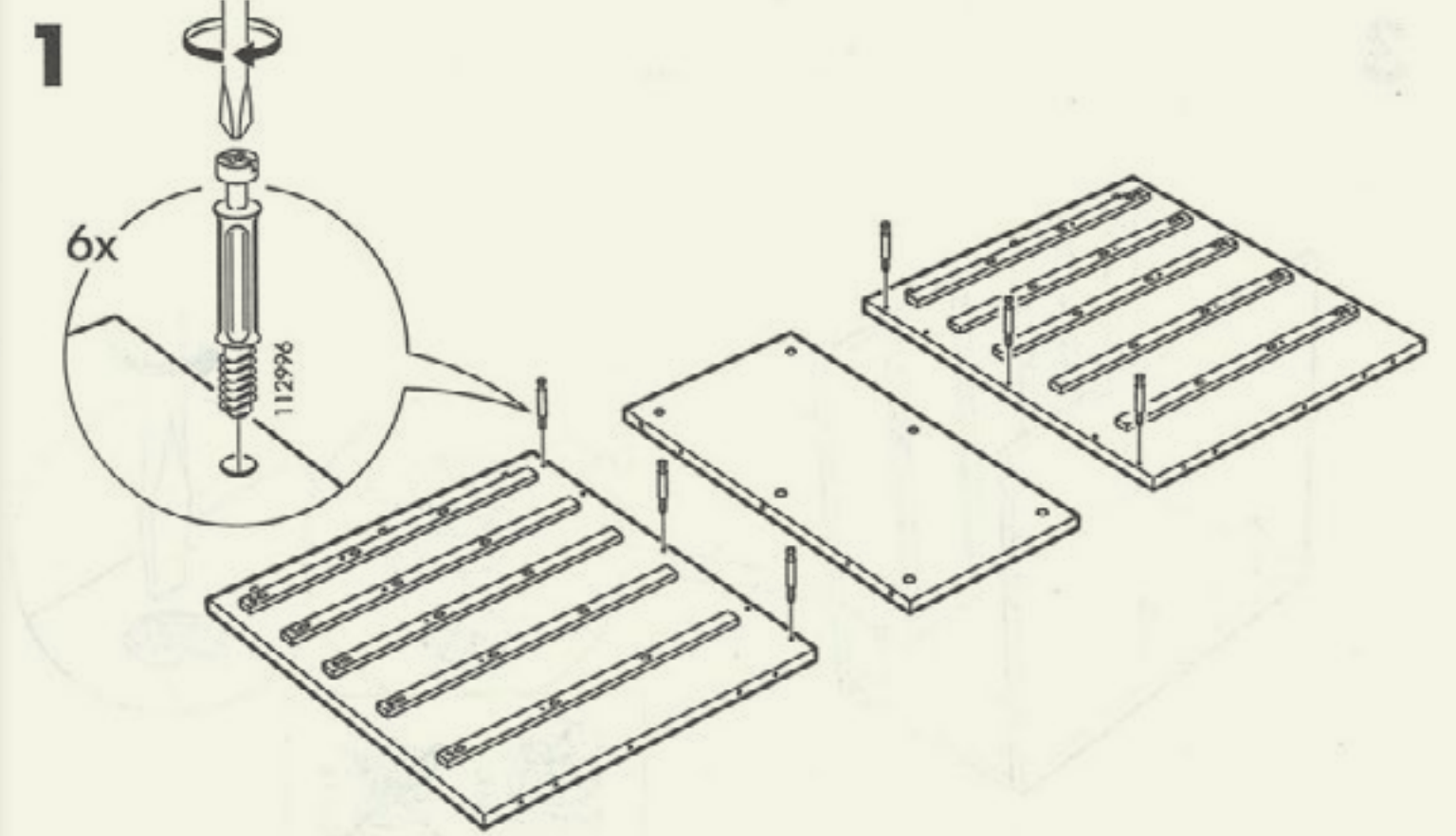


**Netriviální úkol, strmá
učící křivka, výstupy
zprvu nejisté a všechno
trvá strašně dlouho
a je strašně složité.**

**Tím se zabývat
nebudeme,
není k tomu
prostor.**



R PROCESSING



a **R**

**R je programovací
jazyk a prostředí pro
statistické výpočty,
manipulaci s daty
a jejich grafické
znázornění.**

```

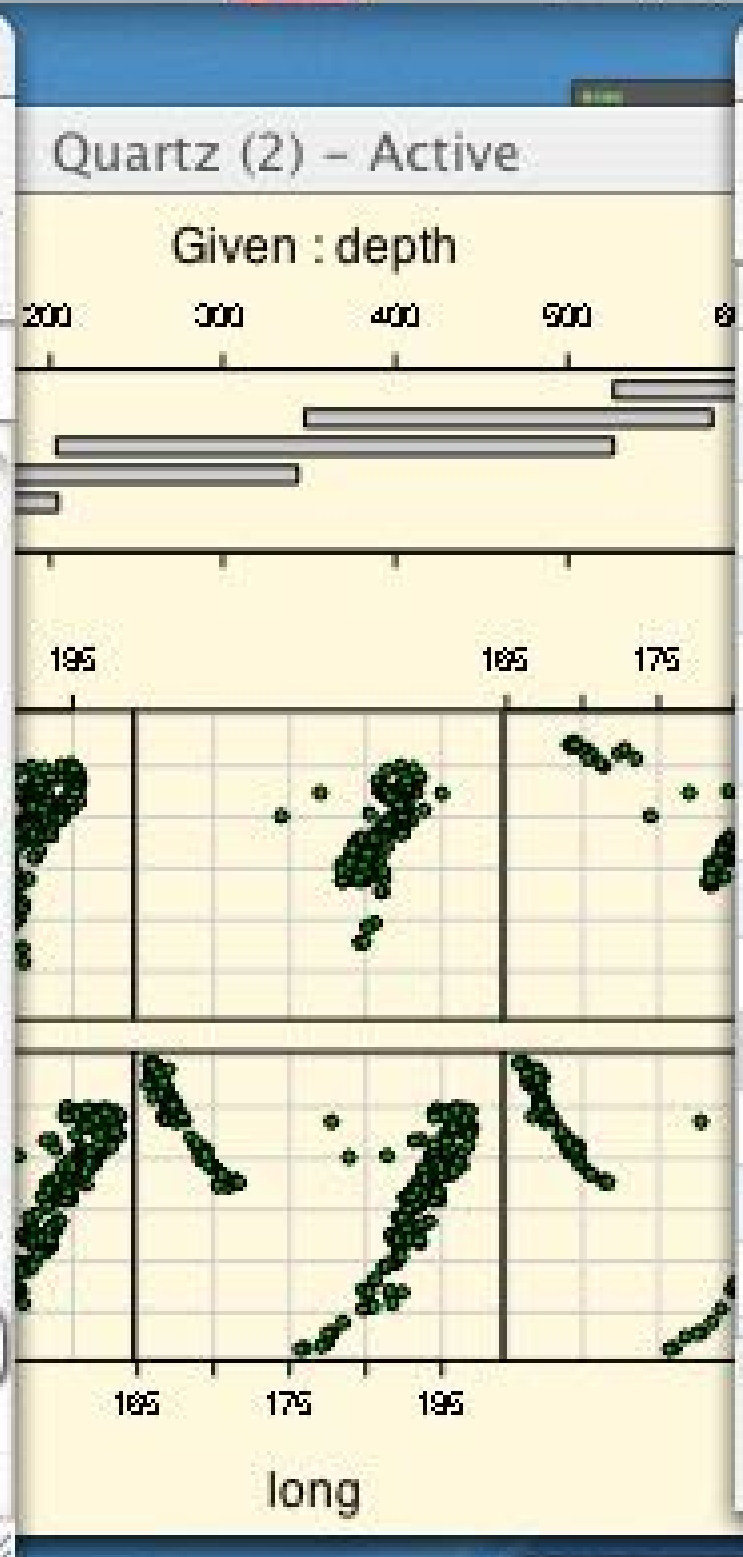
R Console
/Users/jago

rgl.sr> ylen <- ylim[2] - ylim[1] + 1
rgl.sr> colorlut <- terrain.colors(ylen)
rgl.sr> col <- colorlut[y - ylim[1] + 1]
rgl.sr> rgl.clear()
rgl.sr> rgl.surface(x, z, y, color = col)

```

R Data Editor

height	weight
58	115
59	117
60	120
61	123
62	126
63	129
64	132
65	135
66	139
67	142
68	146
69	150
70	154
71	159
72	164



R Workspace Browser

Object	Type	Structure
▶ dati	data.frame	dim: 20 4
g	factor	levels: 10
l	numeric	length: 12
n	numeric	length: 1
▶ opar	list	length: 2
pie.sales	numeric	length: 6
pin	numeric	length: 2
scale	numeric	length: 1
usr	numeric	length: 4
▼ women	data.frame	dim: 15 2
height	numeric	length: 15
weight	numeric	length: 15
x	numeric	length: 87

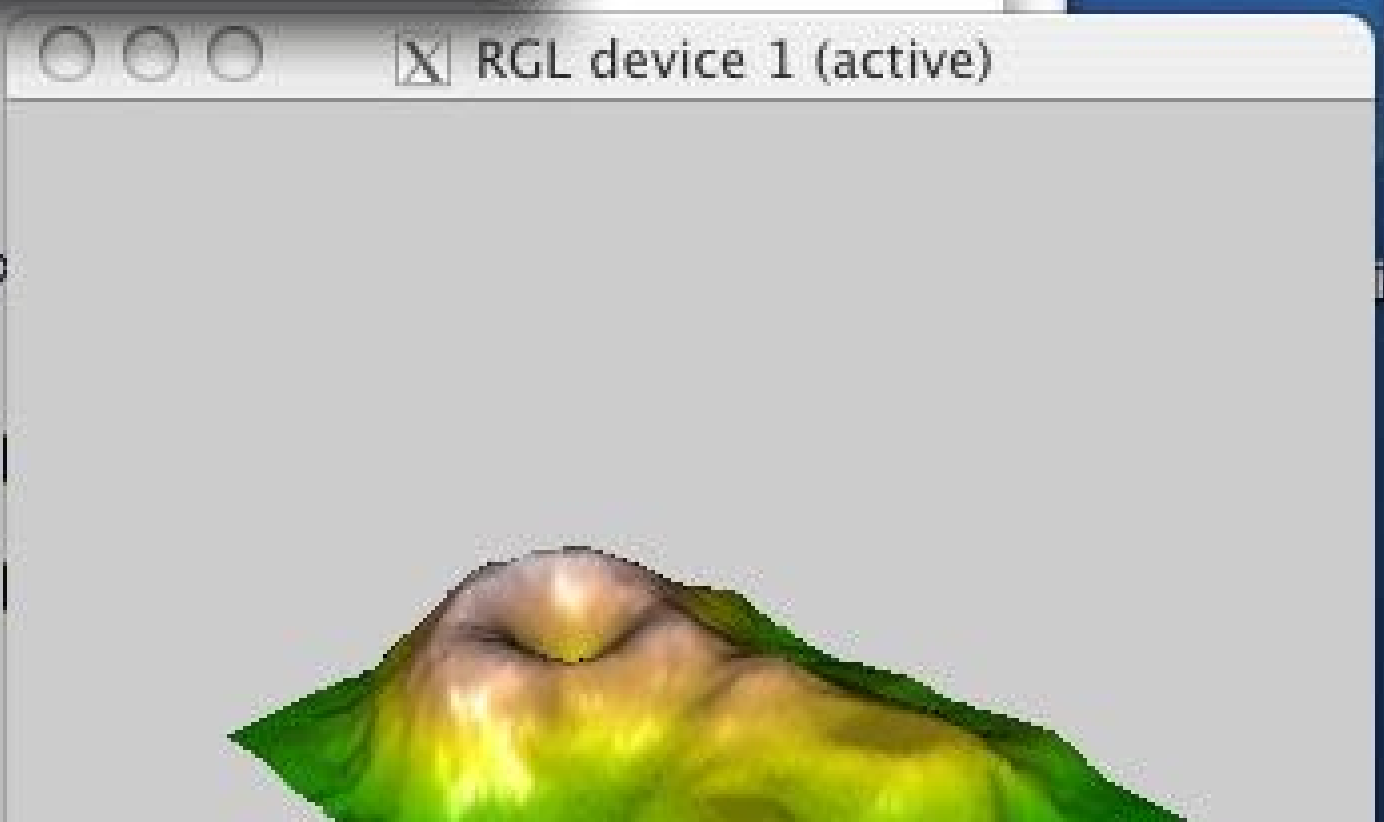
Refresh List

```

BoxDens=function(data, npts = 200., x = c(0.,
add = TRUE, col = 11., border=FALSE,collin
{
dens <- density(data, n = npts)
dx <- dens$x
dy <- dens$y
if(add == FALSE)
plot(0., 0., axes = F, main = "", xlim = x, ylim = y,
ylab = "")
if(orientation == "paysage") {
dx2 <- (dx - min(dx))/(max(dx) - min(dx)) * (x[2.] - x
x[1.]
dy2 <- (dy - min(dy))/(max(dy) - min(dy)) * (y[2.] - y
y[1.]
seqbelow <- rep(y[1.], length(dx))
if(Fill == T)

```


"paysage",



R Package Manager

Refresh List

status	Package	Description
<input checked="" type="checkbox"/> loaded	graphics	The R Graphics Package
<input type="checkbox"/> not loaded	grid	The Grid Graphics Package
<input type="checkbox"/> not loaded	lattice	Lattice Graphics
<input checked="" type="checkbox"/> loaded	methods	Formal Methods and Classes
<input type="checkbox"/> not loaded	mgcv	GAMs with CCV smoothness estimation

The R Graphics Package 

Glyph - a from antikvalt

ISO 8859-2 Latin 2 (Central Europe)

Editing Layers

- All layers
- Outline
- Guidelines
- Hints
- Mask
- Grid
- Vertical metrics
- Global mask
- Glyph metrics
- Components
- Anchors and carets
- Alignment zones
- Shape groups
- Neighbors
- Background

Glyph Properties

Font: antikvalt

NAME: a

UNICODE: 0061

OT: Unassigned 1

20 2 0

32 5 377

235 78 68

(32, -10) (372, 418)

Classes... Apply

Font - antikvalt [F:\...\antikvalt\antikvalt-2.vfb]

ISO 8859-2 Latin 2 (Central Europe)

dollar	percent	ampersand	quotering	parenleft	parenright	asterisk	plus	comma
\$	%	&	‘	()	*	+	,
D	E	F	G	H	I	J	K	L
d	e	f	g	h	i	j	k	l
...
currency	Lcaron	Sacute	section	dieresis	Scaron	Scomma	Tcaron	Zacute
¤	Ĺ	Ś	§	¨	Š	Ș	Ť	Ż
Adieresis	Lacute	Cacute	Ccedilla	Ccaron	Eacute	Eogonek	Edieresis	Ecaron
Ä	Ł	Ć	Ç	Č	É	Ę	Ě	Ě

umfl	abreve	adieresis	lcaron
â	ă	ä	ł
CR	dotlessi	ring	corone
ı	°		
small	F.small	G.small	H.small
E	F	G	H
small	R.small	S.small	T.small
š	š	š	š

OpenType

```

feature liga {
  sub f f i by ffi;
  sub f f l by ffl;
  sub f f by ff;
  sub f i by fi;
  sub f l by fl;
  sub longs l by longs_l;
  sub longs h by longs_h;
  sub longs t by longs_t;
} liga;

```

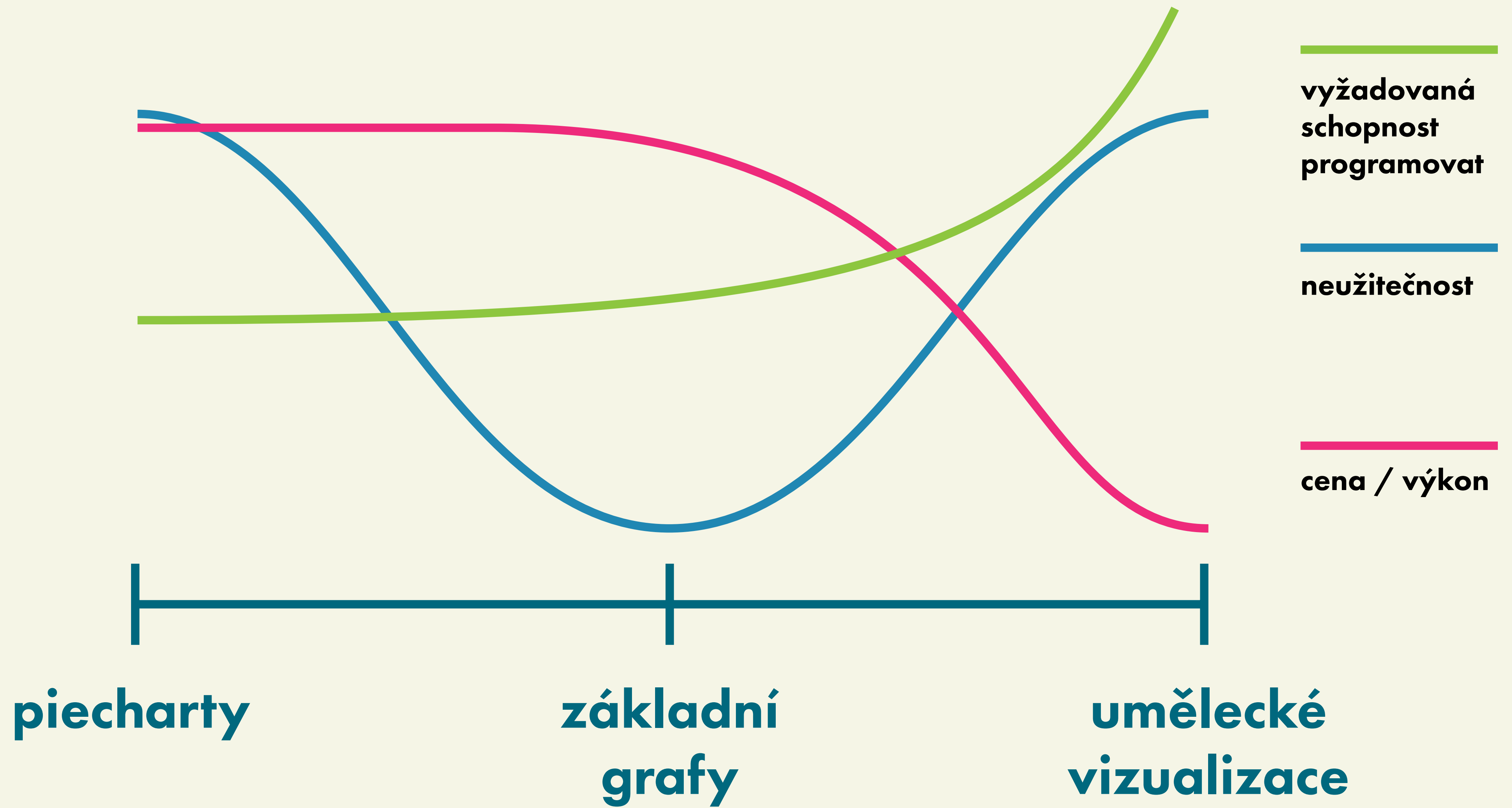
Preview/Metrics - antikvalt

N	zcaron	space	h	roaron	soaron	iacute	n	eacute	space	s	a
397	262	487	321	333	240	486	418	262	328	377	
24		23	31	49	34	21	35		43	32	
18		13	12	29	3	12	38		30	5	

ny

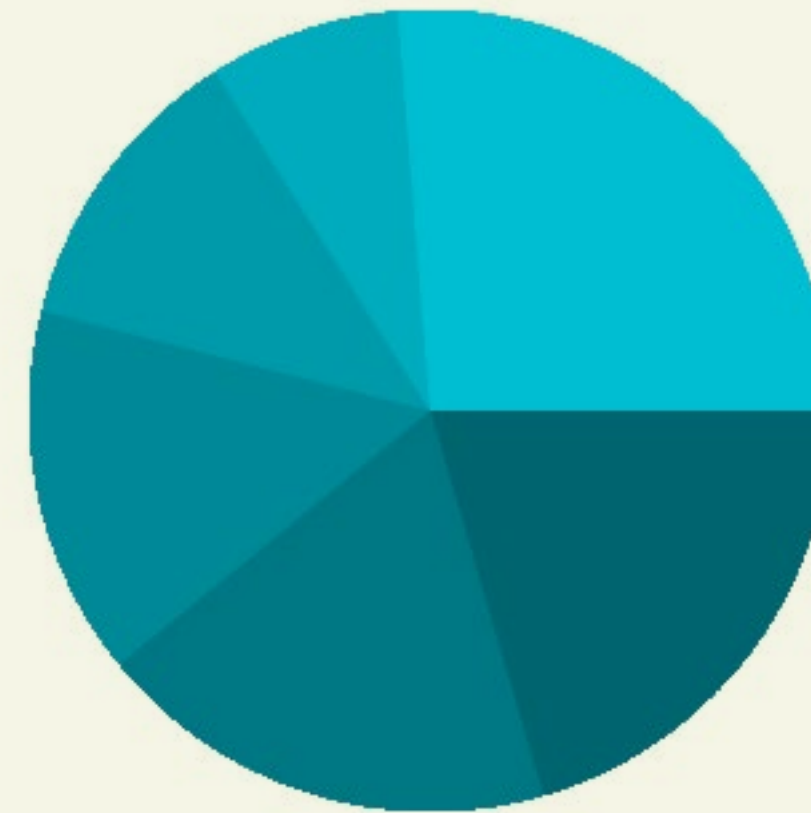
b processing

**Processing je
nadstavba (nejen) nad
Javu, která přidává
další úroveň abstrakce
a tím usnadňuje práci
s vizuálem.**



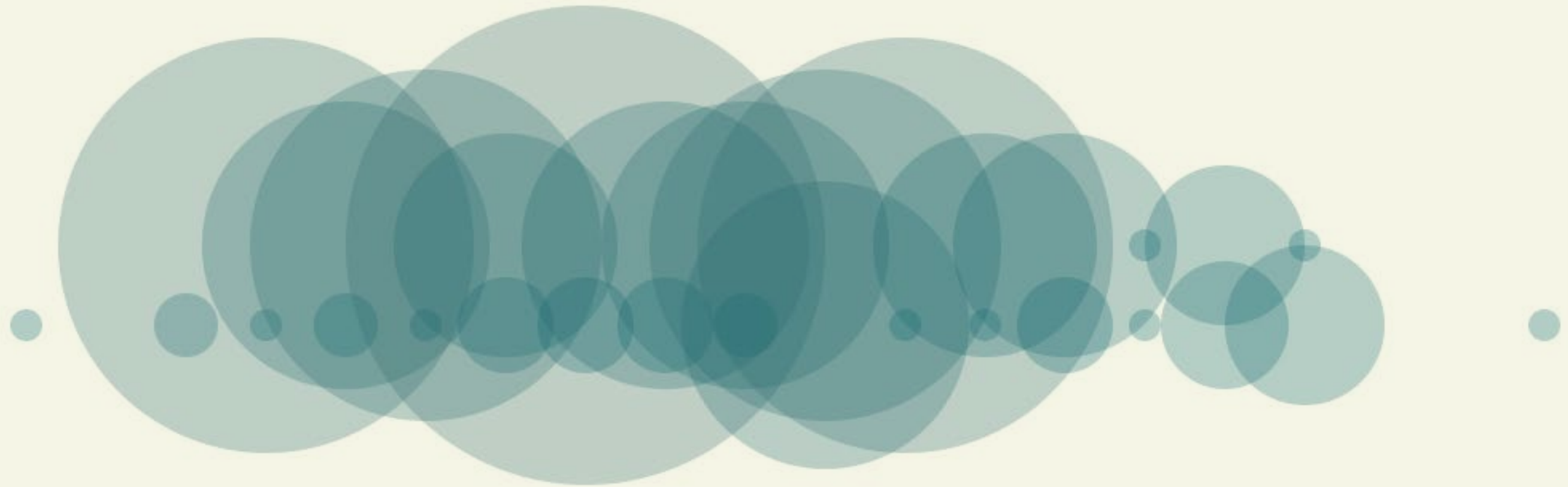
UKÁZKY

Piechart





„Umělecké“ vizualizace



**Processing je
akademická
hračka, se kterou
se dá zvládnout
spousta věcí.
Většinou ale existuje
jednodušší cesta.**



D3.JS

LEAFLET.JS



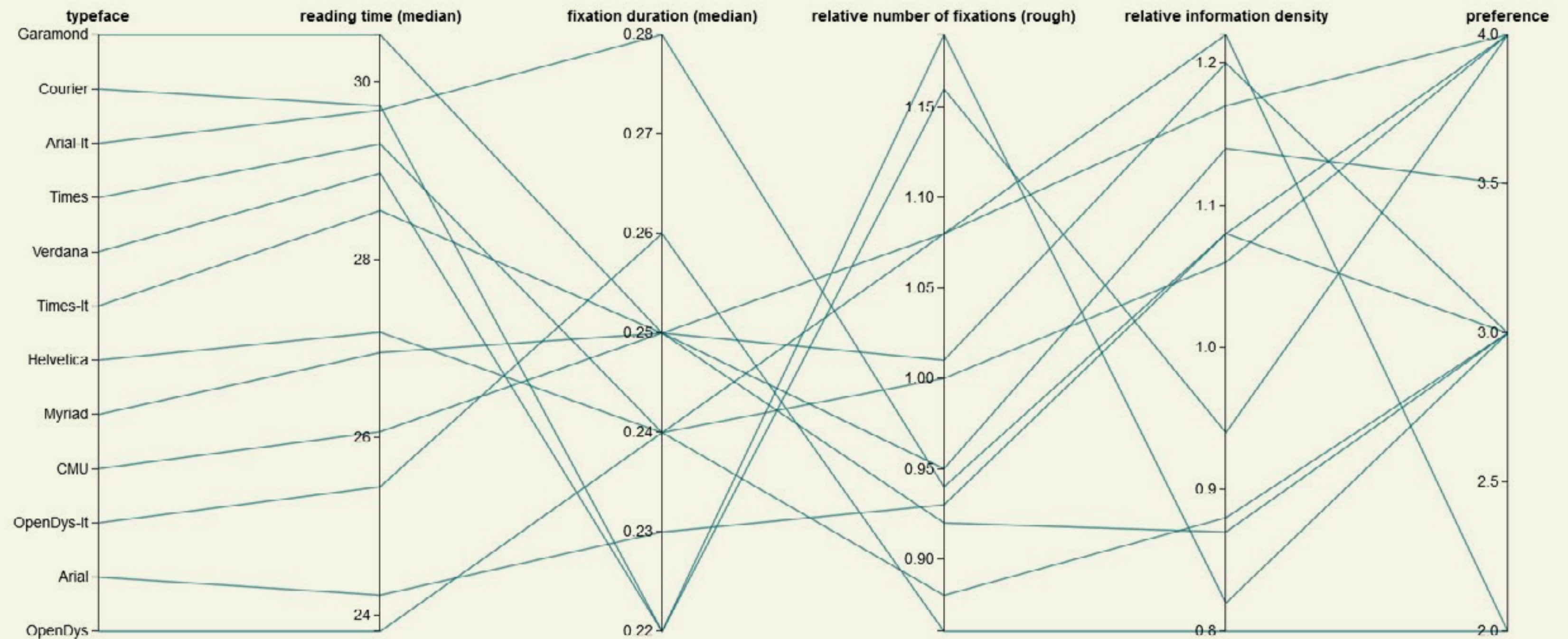
**Stavebnice. Existuje
spoustu předpřipravených
modulů, kterými lze řešit
většinu situací.**

a **d3.js**

**Javascriptová knihovna
pro práci s vizualizacemi.
Někdy krásno, někdy je
potřeba trochu víc rozumět
programování.**

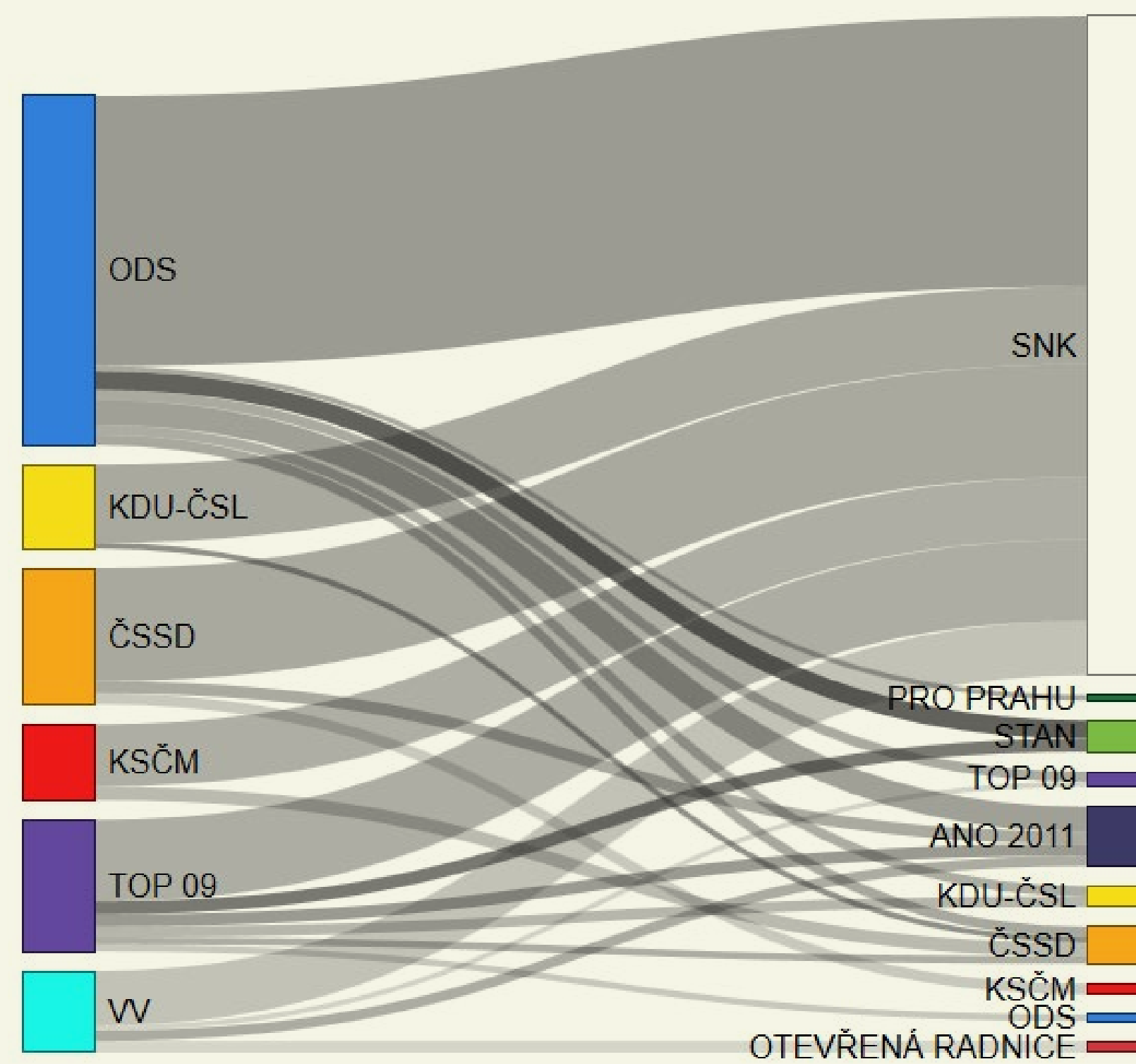


Parallel coordinates





Sankeyův diagram

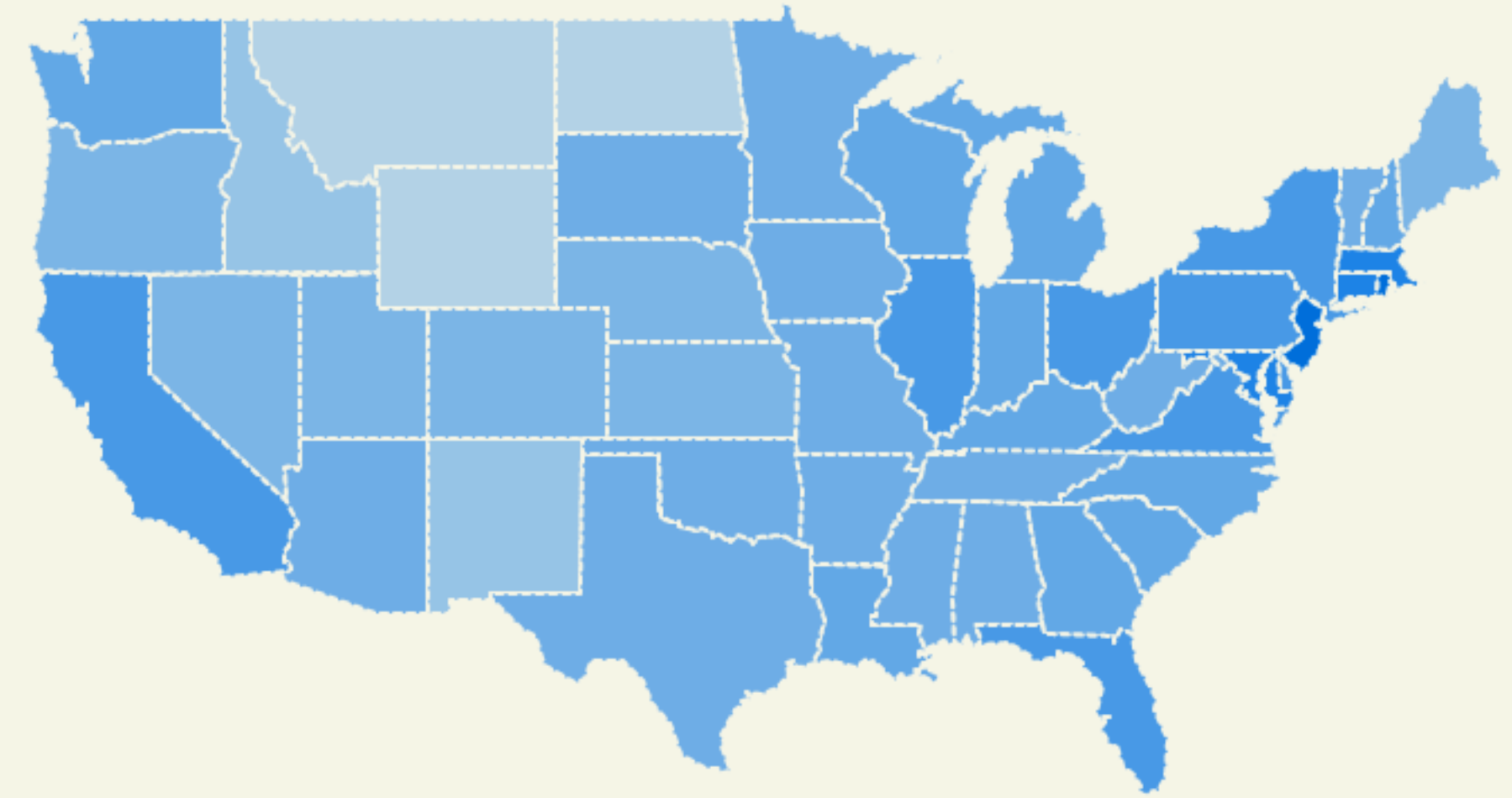


b Leaflet

**Javascriptová knihovna
pro práci s mapovými
vizualizacemi. Velmi dobrá
dokumentace, pěkné
a přehledné tutorialy.**



choropleth



**#04 CO SI
Z DNEŠKA
ODNĚST**

**Programování není
úplná černá magie.**

**Má smysl hrabat se
v cizím kódu.**

#05 ÚKOL

**Máte data ke svým vizualizacím.
Pomocí nástrojů, které jsme si
dneska představili vytvořte jejich
interaktivní vizualizaci. V případě
dotazů pište nebo se 18.12. stavte
na konzultaci.**