

# **PxEdit to GitHub**

Refurbishing a Seasoned APL Application to Open Source

(my concept of a plan)

Veli-Matti Jantunen, 18/9/2024, Glasgow

### Statistics Finland in figures



**Statistics Finland** 

**777** experts, of whom

130 statistical interviewers



**3.70** personnel job satisfaction index, on a scale of 1-5



#### **Funding**

### **EUR 53.9 million**

from the state budget



income from charged services





statistical releases

**160** statistics



**398** user licences **351** data deliveries

for researchers





service deliveries subject to charge

tables in the database

#### **Data needed for statistics**



**95%** from registers 5% with inquiries

95.5% of inquiries could be answered online

### 22222

### 84%

of Finns know Statistics Finland **83%** 

of Finns consider Statistics Finland reliable

5 300

contacts answered by the information service

**17 800** 

page downloads on the stat.fi website

Read more: stat.fi



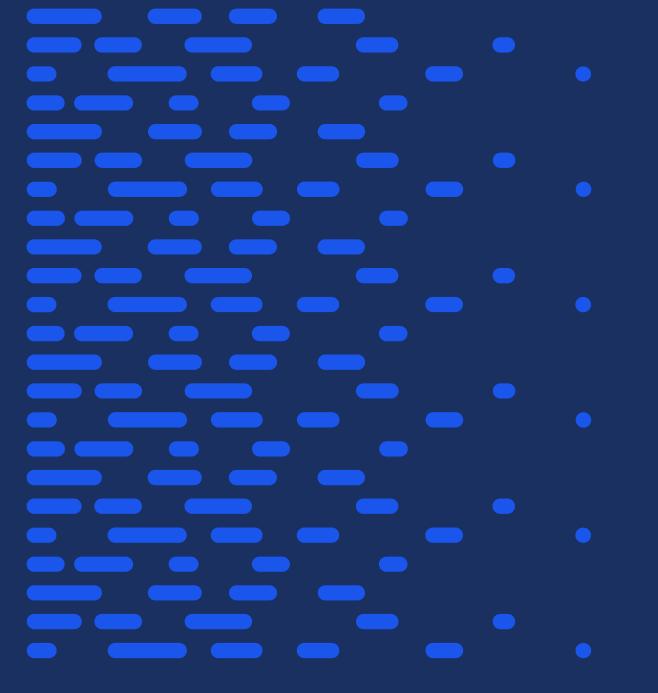


### Statistical inquiries

**137 000** responses from organisations

**202 000** responses from persons and households





Part 0:

Basics

# Some px terminology

#### PxWeb

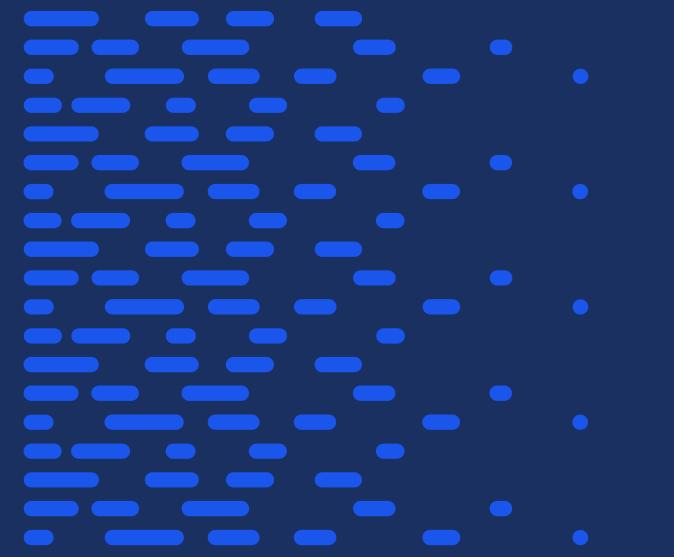
- web server application for **disseminating** statistical tables
- **easy** to use, install and maintain ("made by statisticians for statisticians")
- developed by SCB (Statistics Sweden)
- active developing in several countries
- open source
- uses either relational database or **file-based** collection of tables

#### PX files

- contain both metadata and data
- metadata is given as 'keyword=value' pairs
- data part is stored in **cubic** format
- practically **text files** without size limitation
- most of the statistical tables are easy to convert to px format







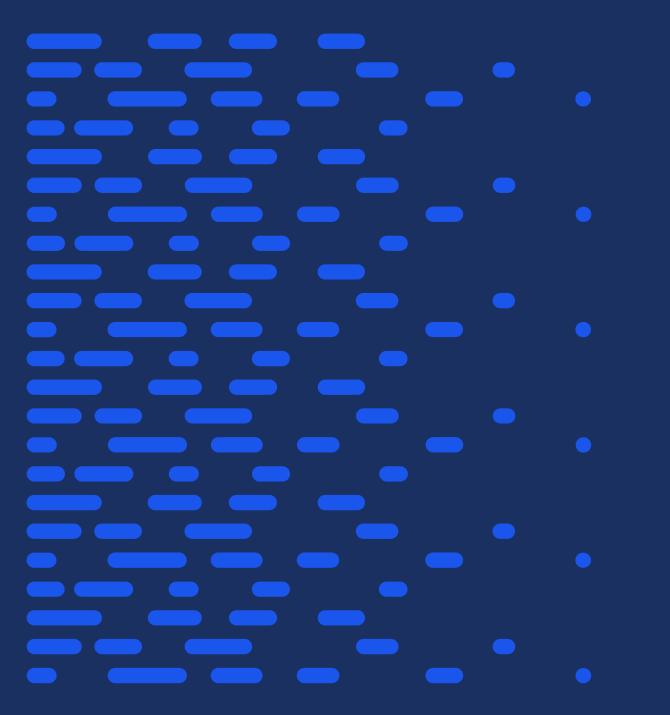
Part 1:
Birth

# Waiting for the end of the (digital) world: Y2K

- Two APL based mainframe databases: ASTIKA (time series) and ALTIKA (regional)
  - I made a *temporary* client-server application for *ASTIKA* (it was in use almost 10 years)
  - Kimmo *Linna* made the first version of PxEdit for handling "huge" *ALTIKA* tables for PxWeb
  - Kimmo left the office.. and Statistics Slovenia started using PxEdit
  - .. suddenly, I was **deeply** involved with another **temporary** application
- PxEdit's usage (and user base) grew quickly
  - no actual competition ©
  - different environments → as pure APL as possible
  - feature requests → continuous development
- Some milestones
  - memory handling
  - change to **Unicode** (getting rid of the *Font Hell*)
  - multilingual tables
- There never seemed to be time to really sit down and redesign







Part 2:
A quarter of a century

later



## **During these years**

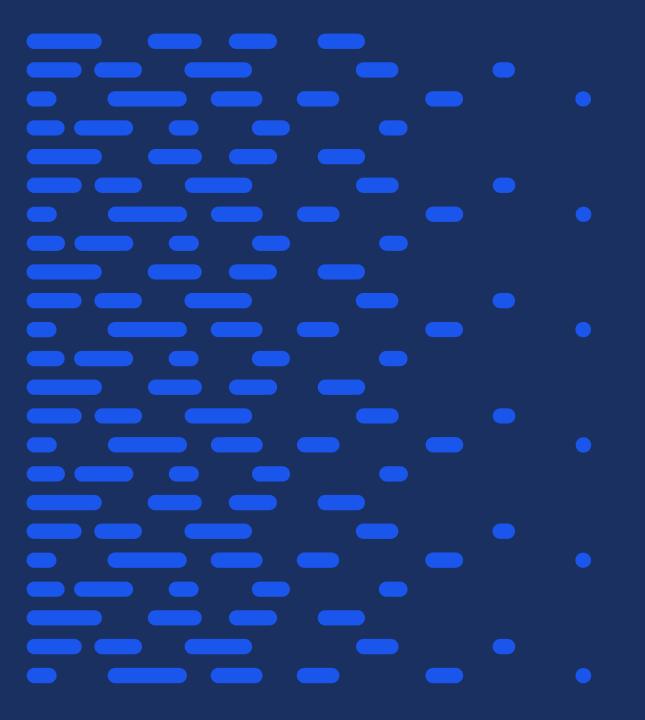
- PxEdit has become de facto production tool in the PX world
  - practically in use in every organisation having PxWeb
  - eager and active user base
- Most important features
  - **structural** table format for moving tables from different environments
  - metadata enriching
  - database features
  - **PxJob** for process automation
- Consulting missions, webinars & other support
  - FAO/CountryStat: Bhutan, Mali, Djibouti, Jordan
  - Slovenia, North Macedonia, Aland Islands, The Nordic Council (Copenhagen)
  - Somalia, Grande Région, FAO/Myanmar
  - Liechtenstein, Uruguay, Beijing, Kosovo, ...
- Annual Px User Meetings



### **Under the bonnet**

- Approximately over 700 functions and operators, over 31 000 lines of code
  - lots of different coding styles
  - hard-coded object names 🕾
  - comments in Finnish ©
- Hundreds of global and semiglobal variables
  - personal settings, environment variables, temporary stuff...
  - documented **somewhere** in the code
  - short and obscure names (even I have problems)
- Long functions with plethora of local variables
  - one- or two-character long names
  - reused widely during the process → meaning and type may vary on the fly
- Most of the stuff lies in the workspace root level
- → It *may* be hard for an outsider to grasp the underlying logic





**Part 3:** 

New Hope

### Covid-19

- The pandemic gave time for thinking and exploring
  - I tested a lot of different approaches
- Step-by-step approach instead of one giant step
  - each step will have certain goals, but not fixed ones
  - each step ends with thorough testing
- Lots of code changes
  - the source code needs to be reorganised, fluffed and rewritten without losing the functionality
  - unnecessary or outdated parts will be removed
  - **new** Dyalog features can be implemented (e.g. *Array Notation*)
- Vision: in the future, a new person having DyalogAPL could...
  - download the latest PxEdit package from GitHub
  - read the short documentation about installation (hopefully there's one)
  - copy the source code in a separate directory
  - copy the accompanying user commands
  - ..and start exploring/fixing



## Improving readability

- Concise object naming is perhaps the most important issue
  - the function names should give a good idea what it is supposed to do CompareLanguage, InjectVariables, CheckTableMeta
  - the variable names should be logical
  - most of the global names will be collected in their own appropriate namespaces CH.dotcodes, lang.default, sys.bits64
  - consistent naming conventions (namespaces/variables/functions..)
- Namespaces make it easier to manage and understand the logic file.Create, dir.Create, space.Create
- Big functions must be chopped to smaller (but logical) ones
  - using **namespaces** instead of semiglobals and/or long argument lists
  - splitting long lines...
- Commenting
  - two commenting strategies: Erkki Juvonen vs Paul Mansour



# **Programming environment**

- Killing the old darlings
  - getting rid of **□SE** based utilities
    - either to User Commands or to the development workspace □SE.uti.fncom → tools.Adjust
  - all the background utilities will be converted as well
    - really old utilities (dating back to 80's)
- Moving from workspace to text files (]LINK)
  - perhaps some kind of specified version control
- Unit testing modules

When everything is ready, I might be able to retire ©

