

Write your own tiny programming system(s)

Why such a strange course topic?

Tomas Petricek, Charles University

✉ tomas@tomasp.net

✈ [@tomasp.net](https://tomasp.net)

🌐 <https://tomasp.net>

🌐 <https://d3s.mff.cuni.cz/teaching/nprg077>



Research

Understanding & improving programming

-  Programming languages, types and theory
-  Interactive programming environments
-  History and philosophy of computing
-  Building tiny systems throughout!

Many different tiny systems!

 PhD, University of Cambridge
Context-aware programming languages

 Microsoft Research Cambridge
F# and applied functional programming

 The Alan Turing Institute, London
Expert and non-expert tools for data science

 University of Kent, Canterbury
Principles of programming systems

 Charles University, Prague
Human-computer interaction and history

Coeffects: Context-aware programming languages

[Home](#)[Theory](#)[Papers](#)

Coeffects are [Tomas Petricek's](#) PhD research project. They are a programming language abstraction for understanding how programs access the *context* or *environment* in which they execute.

The context may be resources on your mobile phone (battery, GPS location or a network printer), IoT devices in a physical neighborhood or historical stock prices. By understanding the neighborhood or history, a *context-aware* programming language can catch bugs earlier and run more efficiently.

This page is an interactive tutorial that shows a prototype implementation of coeffects in a browser. You can play with two simple context-aware languages, see how the type checking works and how context-aware programs run.

This page is also an experiment in presenting programming language research. It is a live environment where you can play with the theory using the power of new media, rather than staring at a dead pieces of wood (although we [have those too](#)).

Short is good! You can always come back.

I'm practical! Show me more examples.

Love theory! Give me all the equations.

Show me all! Time is not an issue.

Programming languages evolve to reflect the changes in the computing ecosystem. The next big challenge for programming language designers is building languages that understand the *context in which programs run*.

This challenge is not easy to see. We are so used to working with context using the current cumbersome methods that we do not even see *that there is an issue*. We also do not realize that many programming features related to *context* can be captured by a simple *unified abstraction*. This is what coeffects do!

What are some examples of context-aware computations?

- In cross-platform code, the functions available on different platforms are a context. You can use `#if`. If you get this wrong, your code won't even compile!
- In the [Game of Life](#) or [weather simulations](#), each cell in a grid accesses neighboring cells. But do you know how many neighbors it needs?

[➔ Read more](#)

What problem are coeffects solving?



Demo

Coeffects playground

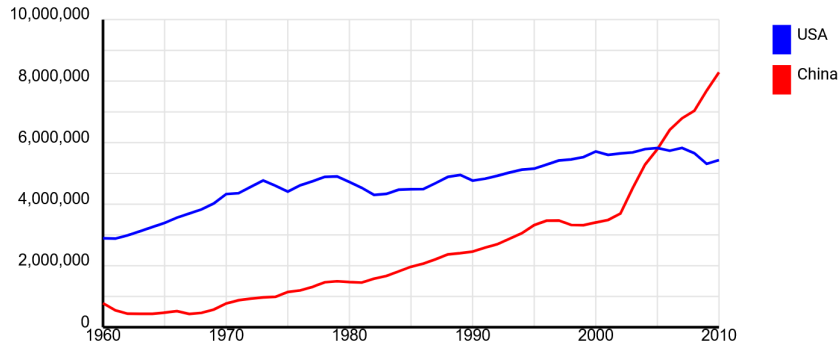
Needed "implementation" to finish my PhD!

How to show potential uses of theoretical work?

Tiny web demos of two potential applications

The Alan Turing Institute Data playground

```
let china = worldbank.byCountry.China.'Climate Change'. 'CO2 emissions (kt)'  
  .setProperties(seriesName="China")  
  
let usa = worldbank.byCountry.'United States'. 'Climate Change'. 'CO2 emissions (kt)'  
  .setProperties(seriesName="USA")  
  
compost.charts.lines([china,usa]).setColors(["red", "blue"])  
  .setAxisX(1960).setLegend("right").setSize(800,300)
```



Demo The Gamma project

Making programmatic data
exploration accessible to
non-programmers

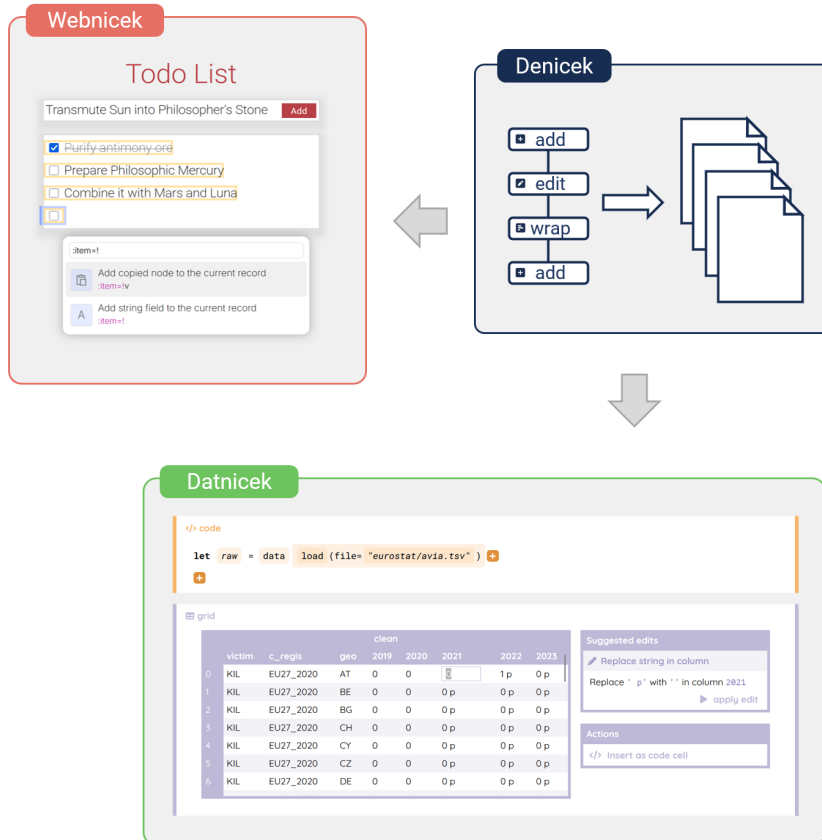
Small typed language, but
interaction is why it works!

Demo

The Denicek project

Computational substrate
for end-user programming

Making implementation of
end-user programming
experiences easier...



Practical details

Course structure

- Videos to watch in advance
- Hands-on 3-hour labs
- Code skeleton with detailed comments



Doing the course

- Six different languages or systems
- Come to the labs to get help!
- Complete basic tasks for 4/6 systems



PRG*PRG / Prague

Programming systems
& languages research

D3S at MFF CUNI

PRL at FIT CTU

Ask us about PhD &
post-doc opportunities!

<https://prgprg.org>