Do You Use Stu? Soon, You'll Do.

Stu Is for You If …

- You've had to submit an old version of a figure in a paper, because you didn't remember how to regenerate it quickly
- You use an old version of your dataset, because regenerating it properly would be too time consuming
- You're afraid of reviewers telling you to "re-run experiments with a larger dataset" because you don't remember how you ran the experiments in the first place
- The person who originally prepared a dataset is now gone and you can't reproduce what they did
- People ask you for old code, but you wouldn't even be able to run it yourself
- You avoid changing a file for fear that your build automation tool will rebuild everything
- Your current project contains code that generates Makefiles automatically

Stu Is a Build Tool

Stu automatises builds. It is used for compiling software, generating Latex papers, and driving small and large data mining projects. Students use it for their work inside and outside WeST. Stu works similarly to Make, but has two killer features: parameters, and dynamic dependencies. Stu was written to be used in KONECT, where it replaced Make, which did not scale. Stu runs in milliseconds instead of minutes. Stu uses kilobytes of memory, not megabytes. Stu syntax is simple yet powerful, expressive yet concise. Stu has a stable API and over 500 unit tests. Stu is compatible with almost all systems as it avoids exotic interfaces. Stu is written in C++11. Stu reports errors like a real programming language, not like a text preprocessor.

Feature 1: Parameters

```
dat/prediction.$METHOD.$NETWORK.mat:
    uni/out.$NETWORK
    m/method_$METHOD.m
{
    ./matlab m/method_$METHOD.m
}
```

Feature 2: Dynamic Dependencies

```
@prediction: [dat/dep.prediction];
dat/dep.prediction: dat/NETWORKS dat/METHODS
{
    for NETWORK in $(cat dat/NETWORKS) ; do
        for METHOD in $(cat dat/METHODS) ; do
            echo dat/prediction.$METHOD.$NETWORK.mat
        done
    done
}
```

Make, Cook, Brew or Stu?

Stu follows a long line of build tools, starting with "make" from 1976. Many subsequent build tools have names related to cooking. Stu follows this tradition and also honours the author of the original Unix Make, Stuart Feldman.

License: GNU

Stu is free software (GPLv3) made for POSIX platforms. Get it, clone it and contribute to it at `https://github.com/kunegis/stu`

WeST People and Knowledge Networks

Jérôme Kunegis
Institute for Web Science and Technologies
University of Koblenz-Landau
kunegis@uni-koblenz.de

UNIVERSITÄT KOBLENZ · LANDAU

Institute for Web Science and Technologies · University of Koblenz-Landau, Germany