

Scalable Network Analysis

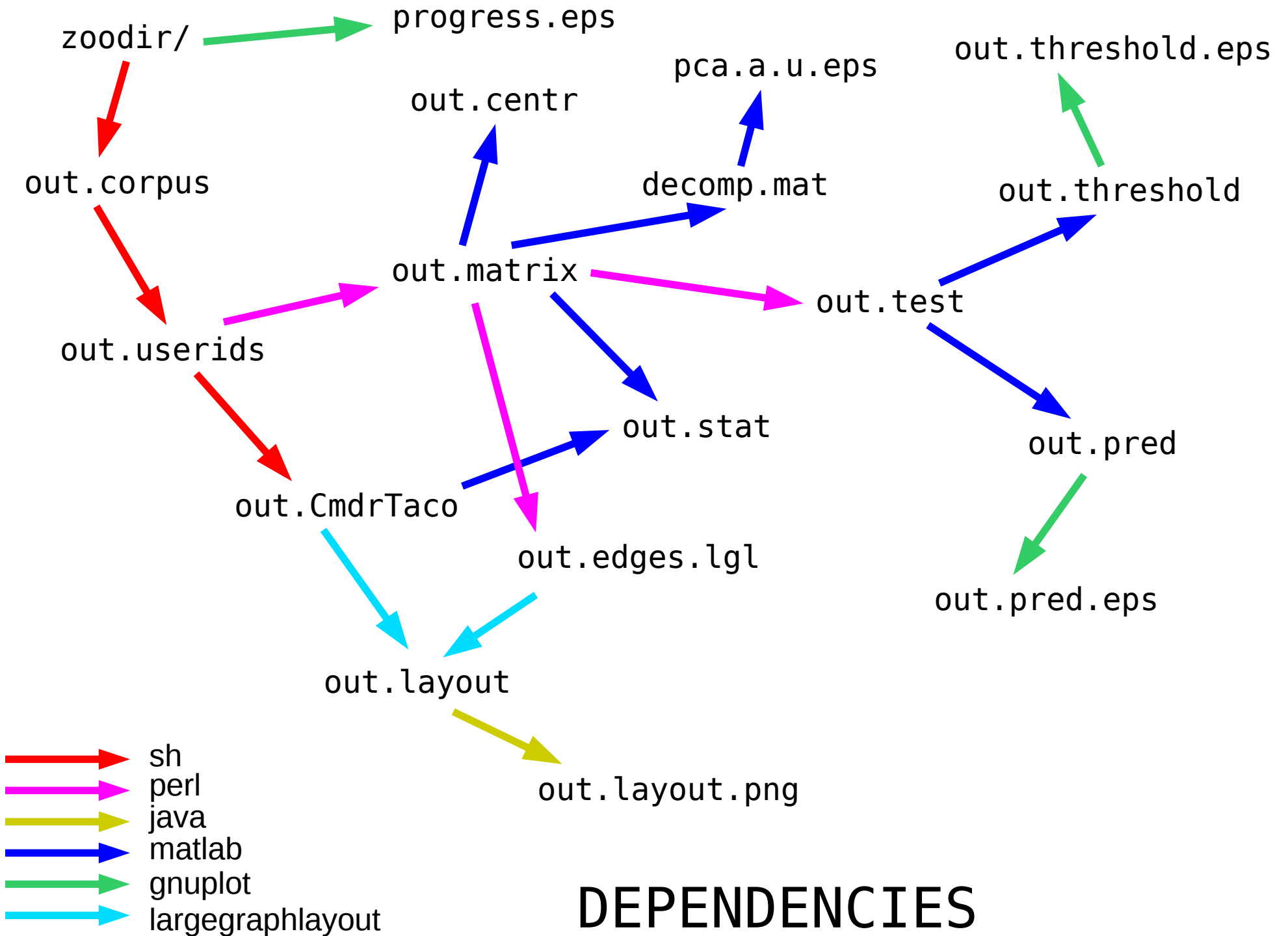
Use Case: Stu

<https://github.com/kunegis/stu>

KONECT Analysis with Stu

- Run tasks in parallel
- Never rebuild a file
- Interrupt at any point

```
SH
USER      PID %CPU    ELAPSED   TIME COMMAND
kunegis  12461 0.0      18:37:30 00:00:01 stu -j 20 -k @network.amazon0302 @network.amazon0312 @network.amazon0505 @networ
-
%CPU      MEM [k]    TIME      LEFT LOG
105       3,714,016 18:58:16  /data/kunegis/tmp/m.lybl.amazon0505.log
101       3,710,372 18:51:04  /data/kunegis/tmp/m.lybl.amazon0312.log
100       3,550,792 18:42:43  /data/kunegis/tmp/m.lybl.amazon0302.log
118       3,578,412 15:49:48  /data/kunegis/tmp/m.lybl.sx-superuser.log
119       3,579,932 14:56:19  /data/kunegis/tmp/m.lybl.sx-askubuntu.log
99        6,630,424 12:27:07  3-14:51:18 /data/kunegis/tmp/m.hopdistr_time_comp.full.sx-askubuntu.log
99        4,089,572 12:25:07  3:42:50 /data/kunegis/tmp/m.statistic_comp.tour4.higgs-twitter-social.log
99        4,089,572 12:05:29  4:16:20 /data/kunegis/tmp/m.statistic_comp.squares.higgs-twitter-social.log
0         4,413,096 3:38     /data/kunegis/tmp/m.layout.higgs-twitter-social.log
```



DEPENDENCIES

...

out.corpus: mkcorpus
./mkcorpus

out.userids: mkuserids out.corpus
./mkuserids

out.matrix: mkmatrix out.corpus out.userids
./mkmatrix

out.CmdrTaco: out.userids mkname
./mkname CmdrTaco

out.stat: out.userids out.corpus out.matrix matlab stat.m out.CmdrTaco
./matlab stat >\$@

out.centr: out.matrix matlab centr.m
./matlab centr >\$@

out.pred: out.test matlab pred.m
./matlab pred >\$@

out.test: ./mksplit out.matrix
./mksplit

out.pred.eps: mkpred out.pred
./mkpred

...

make

Make: with many datasets

```
define TEMPLATE_fit

$(foreach NETWORK, $(NETWORKS), fit.$(1).$(NETWORK)): \
fit.$(1).%: plot/fit.a.$(1).%.eps

dat/fit.$(1).%.mat: \
    dat/info.% dat/decomposition_split.source.$(1).%.mat \
    dat/split.%mat dat/means.%mat m/fit.m
    NETWORK=$$* DECOMPOSITION=$(1) $(OCTAVE) m/fit

plot/fit.a.$(1).%.eps: dat/fit.$(1).%.mat m/fit_plot.m
    NETWORK=$$* DECOMPOSITION=$(1) $(OCTAVE) m/fit_plot

endif
$(foreach DECOMPOSITION, $(DECOMPOSITIONS), $(eval $(call TEMPLATE_fit,$(DECOMPOSITION))))

define TEMPLATE_fit_asym
fit.$(1).all: $(foreach NETWORK, $(NETWORKS_ASYM), fit.$(1).$(NETWORK))
endif
$(foreach DECOMPOSITION, $(DECOMPOSITIONS_ASYM), $(eval $(call TEMPLATE_fit_asym,$(DECOMPOSITION))))

define TEMPLATE_fit_any
fit.$(1).all: $(foreach NETWORK, $(NETWORKS), fit.$(1).$(NETWORK))
endif
$(foreach DECOMPOSITION, $(DECOMPOSITIONS_ANY), $(eval $(call TEMPLATE_fit_any,$(DECOMPOSITION))))

$(foreach NETWORK, $(NETWORKS), fit.all.$(NETWORK)): \
fit.all.%: $(foreach DECOMPOSITION, $(DECOMPOSITIONS_ANY), fit.$(DECOMPOSITION).%)
$(foreach NETWORK, $(NETWORKS_ASYM), fit.all.$(NETWORK)): \
fit.all.%: $(foreach DECOMPOSITION, $(DECOMPOSITIONS_ASYM), fit.$(DECOMPOSITION).%)
```

Stu Script: Example

```
@degcc: @degcc.[dat/NETWORKS_SQUARE];
```

```
@degcc.$network: plot/degcc.$network.eps;
```

```
plot/degcc.$network.eps:  
  m/degcc.m $[-t MATLABPATH]  
  dat/cluscod.$network.mat  
  uni/out.$network  
{  
  ./matlab m/degcc.m  
}
```

<https://github.com/kunegis/stu>

What Is Stu?

- “What Make should be”
- “A declarative programming language where variables are files”
 - cf. Unix: “everything is a file”
- “What if the shell was declarative?”
 - use shell syntax
- Don't hardcode anything
 - no builtin rules; can be written *in* Stu
- Programming language-agnostic
- Scalability for data mining
 - Ctrl-C, signals, precious files, $O(1)$, etc.
- “Make” \Rightarrow “Cook” \Rightarrow “Stew”
- Free Software, GPLv3
- cf. Stu Feldman, inventor of Make

Stu: Information

<https://github.com/kunegis/stu>

For motivation, see:

<https://networkscience.wordpress.com/2017/08/15/an-update-on-the-stu-build-system-stu-2-5/>