



DEPENDENCIES STATIC ANALYSIS

— AND —

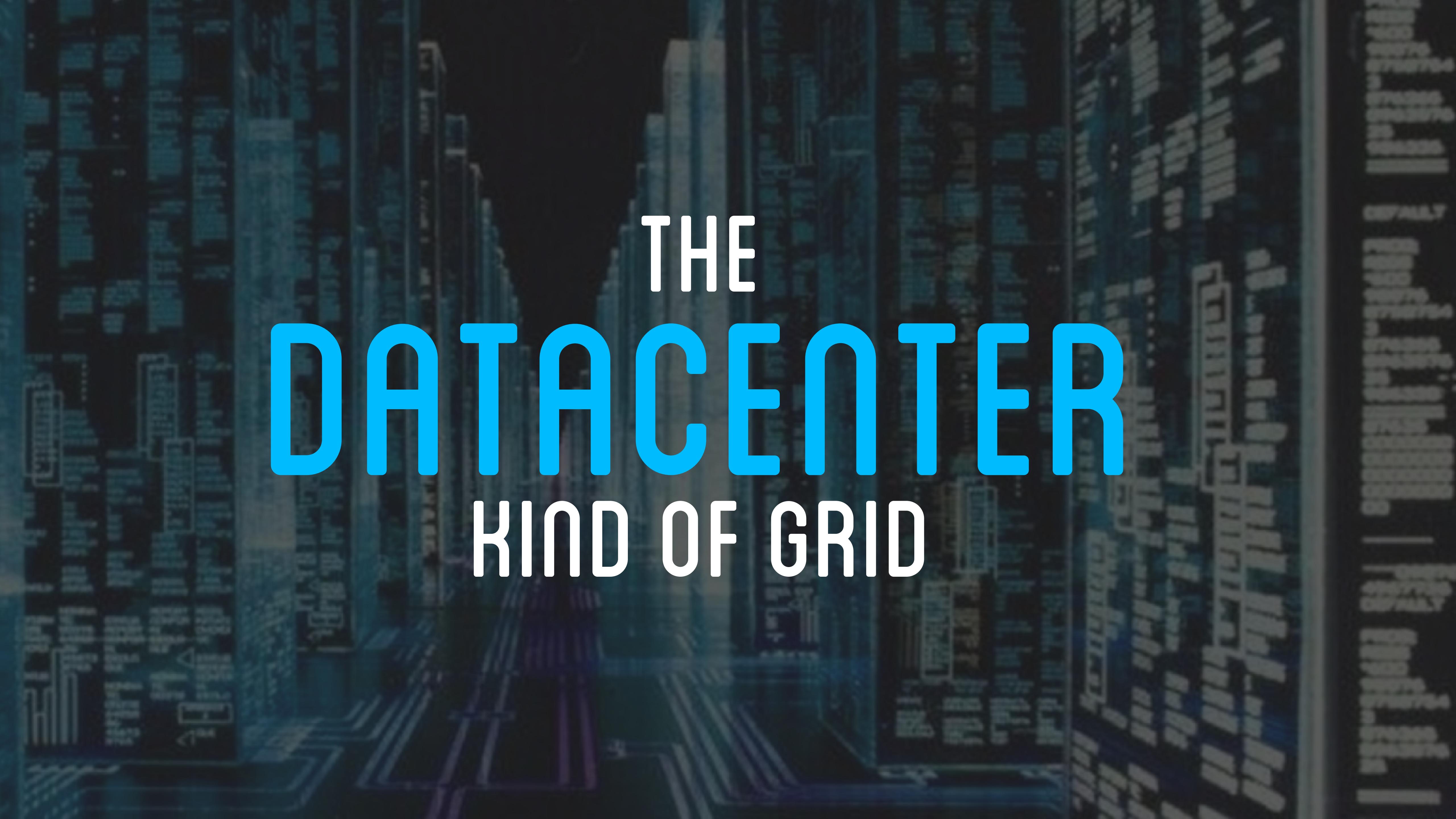
DEPENDENCIES STATIC ANALYSIS — AND — THE GRID



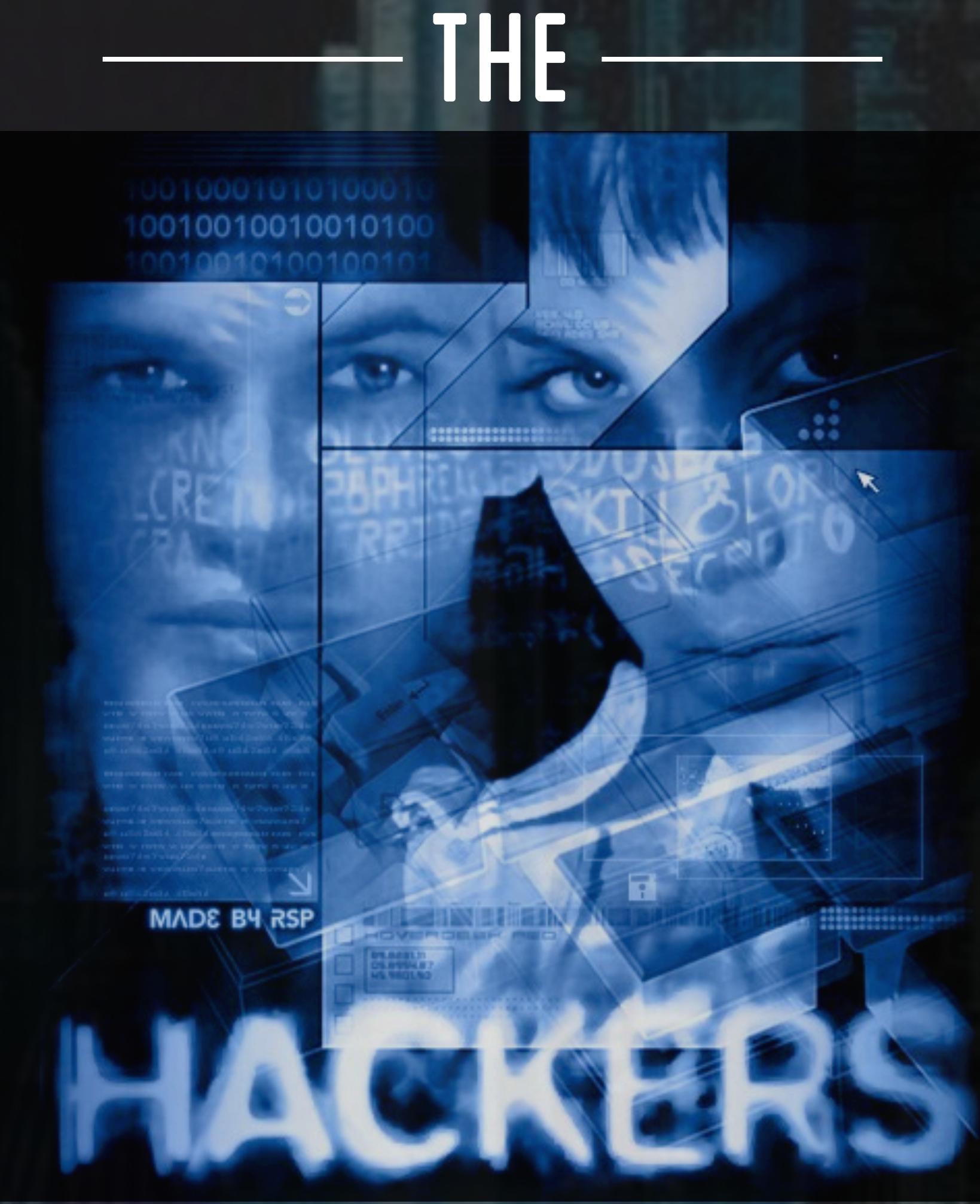
A dark, glowing figure with a hood and glowing blue hands, set against a dark background with glowing blue lines.

NOT THE
TRON
KIND OF GRID





THE DATACENTER KIND OF GRID



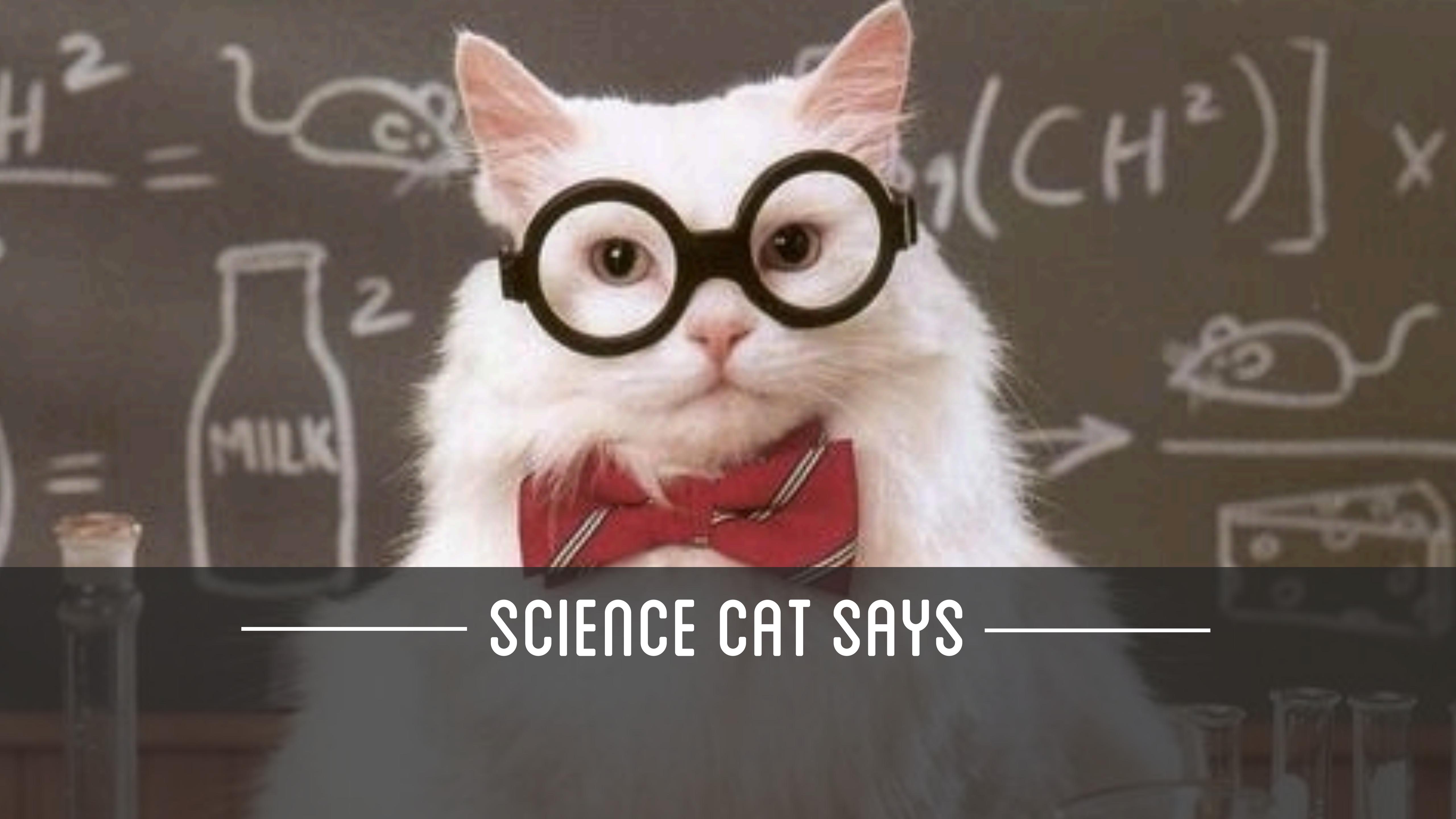
THE
KIND OF GRID





YOU ARE PROBABLY THINKING

BUT WHY?



— SCIENCE CAT SAYS —



— SCIENCE CAT SAYS —
“ GOOD QUESTION ”



A nighttime photograph of a city street. In the foreground, there is a crosswalk with diagonal stripes. In the background, there are several multi-story buildings, some with lit windows. The sky is dark, and streetlights are visible, creating a warm glow.

I LED AN ACADEMIC
RESEARCH STUDY
EARLIER THIS YEAR



COLUMBIA UNIVERSITY
IN THE CITY OF NEW YORK

— SHOW YOUR ALMA MATER —



COLUMBIA UNIVERSITY
IN THE CITY OF NEW YORK

— SOME LOVE —

— SHOW YOUR ALMA MATER —



COLUMBIA UNIVERSITY
IN THE CITY OF NEW YORK

— SOME LOVE —

— AND YES —

Toward decreasing mean decision-time in Open Source with com(STAT)²

Charlie Robbins, Jane Kim, William McAuliff

Columbia University

New York, NY, 10027, USA

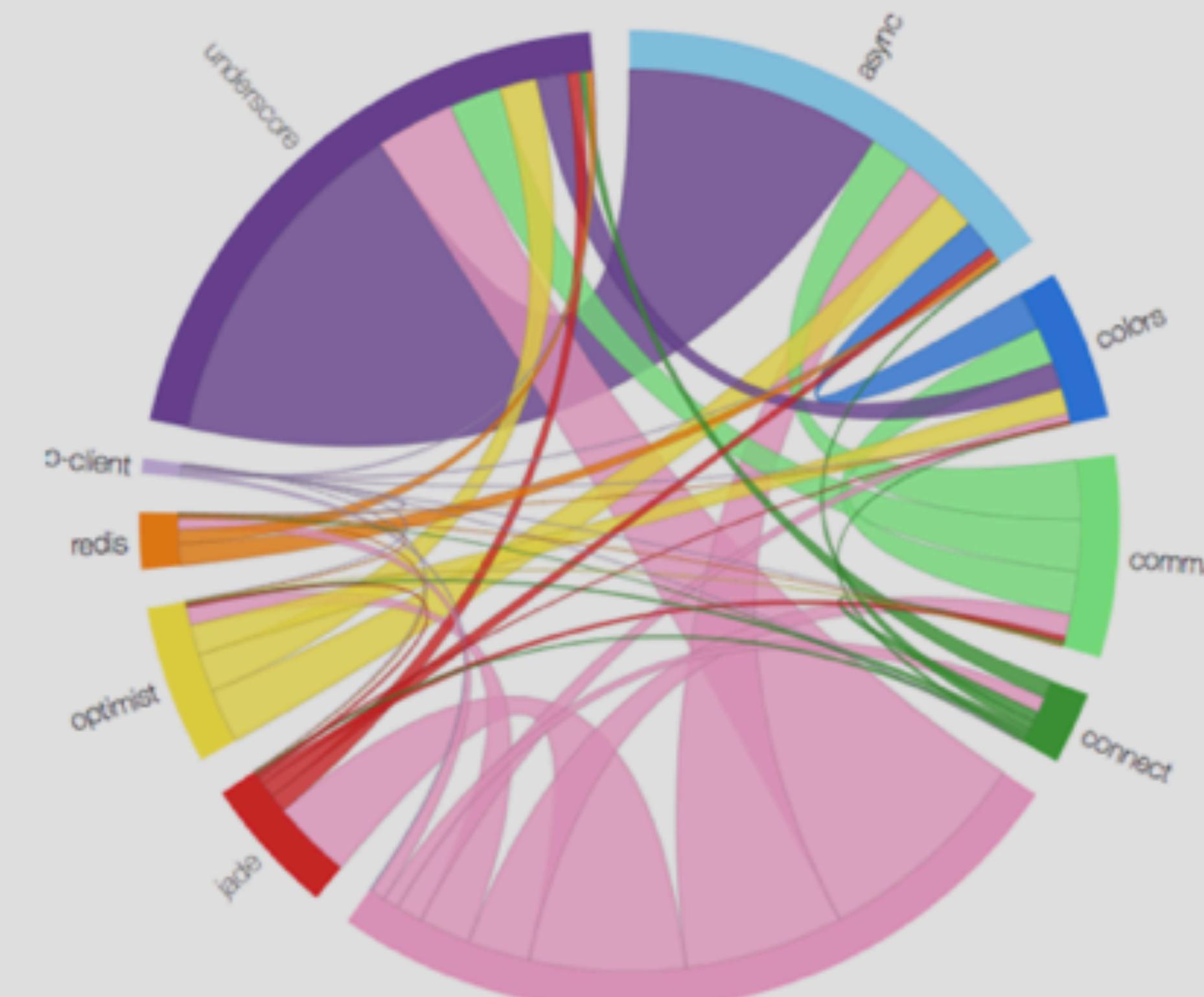
{ cjr2139, jk3316, wom2102 } @ columbia.edu

May 14, 2014

Abstract

Modules, small reusable pieces of code, have long been the goal of sustainable Software Engineering practices. In recent years in some circles, particularly Javascript, this goal has become a reality. Having these large Javascript applications depend on hundreds of small modules presents its own decision-making challenges for module authors. Most are flying completely blind with respect to how their module(s) are being used and what constitutes a “breaking change” for their user base.

This decision-making for mature projects often takes more time than the engineering work itself. This paper presents a COMprehensive STATistical STATIC analysis interface, com(STAT)², that aims to decrease the mean decision time for module authors and evaluates it on well-known authors in the popular Javascript environment of Node.js and npm (Node Packaged Modules).



Toward decreasing mean decision-time in Open Source with $\text{com}(\text{STAT})^2$

Charlie Robbins, Jane Kim, William McAuliffe

Columbia University

New York, NY, 10027, USA

{ cjr2139, jk3316, wom2102 } @ columbia.edu

May 4, 2018

RESEARCH PAPER AND EVERYTHING

Most are flying completely blind with respect to how their modules(s) are being used and what constitutes a “breaking change” for their user base.

This decision-making for mature projects often takes more time than the engineering work itself. This paper presents a **COM**prehensive **STAT**istical **STATIC** analysis interface, *com(STAT)^2*, that aims to decrease the mean decision time for module authors and evaluates it on well-known authors in the popular Javascript environment of *Node.js* and *npm* (Node Packaged Modules).

AND EVERYTHING





A fluffy, light-colored dog with a dark brown patch on its back is wearing black-rimmed glasses and a red patterned tie. It is looking slightly to the left. The background is a dark, out-of-focus image of the dog's face.

ACADEMIC HCI
IS A LITTLE SILLY



ACADEMIC HCI IS A LITTLE SILLY

WHY DO LIKERT SCALES HAVE A NAME?





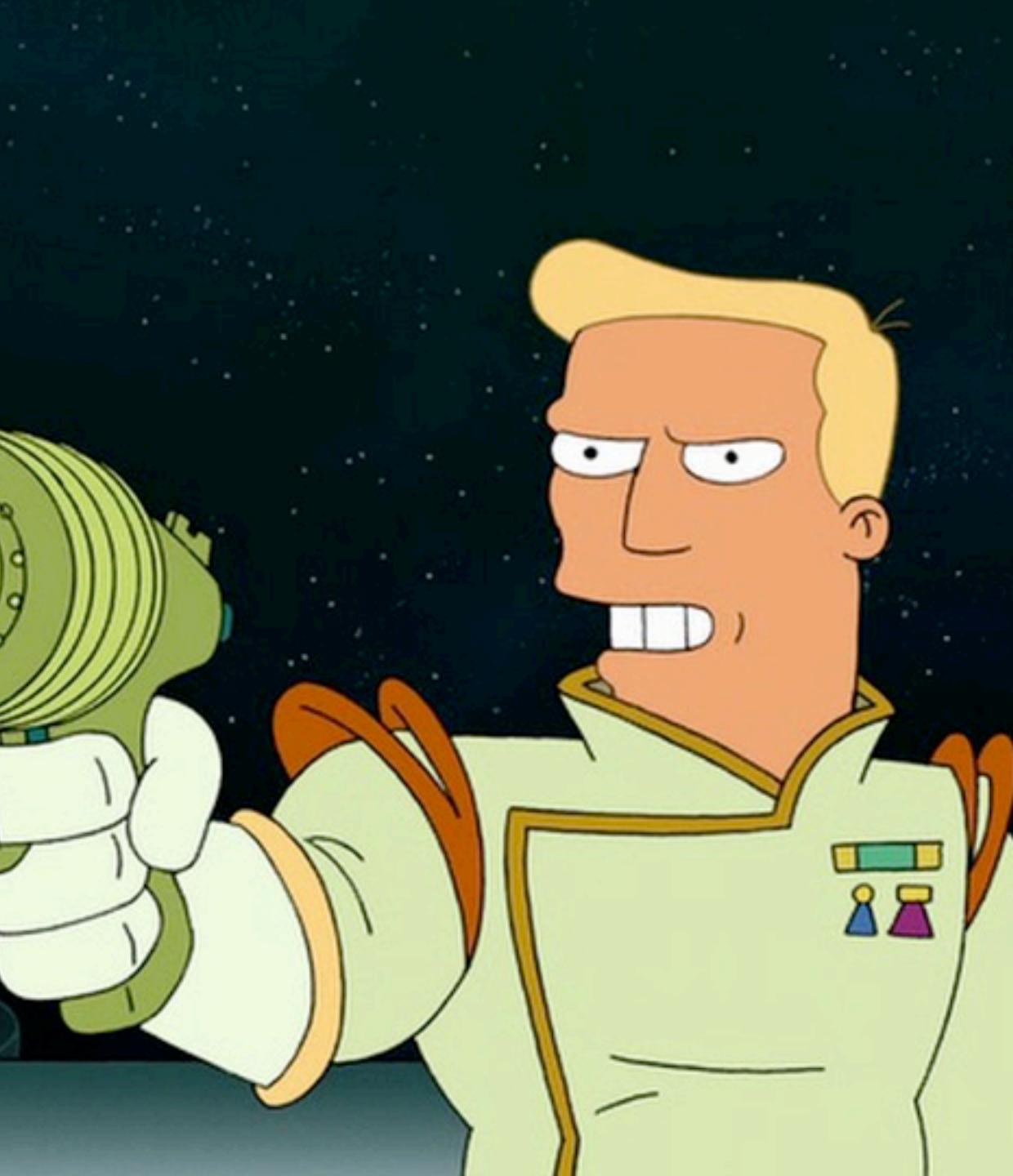
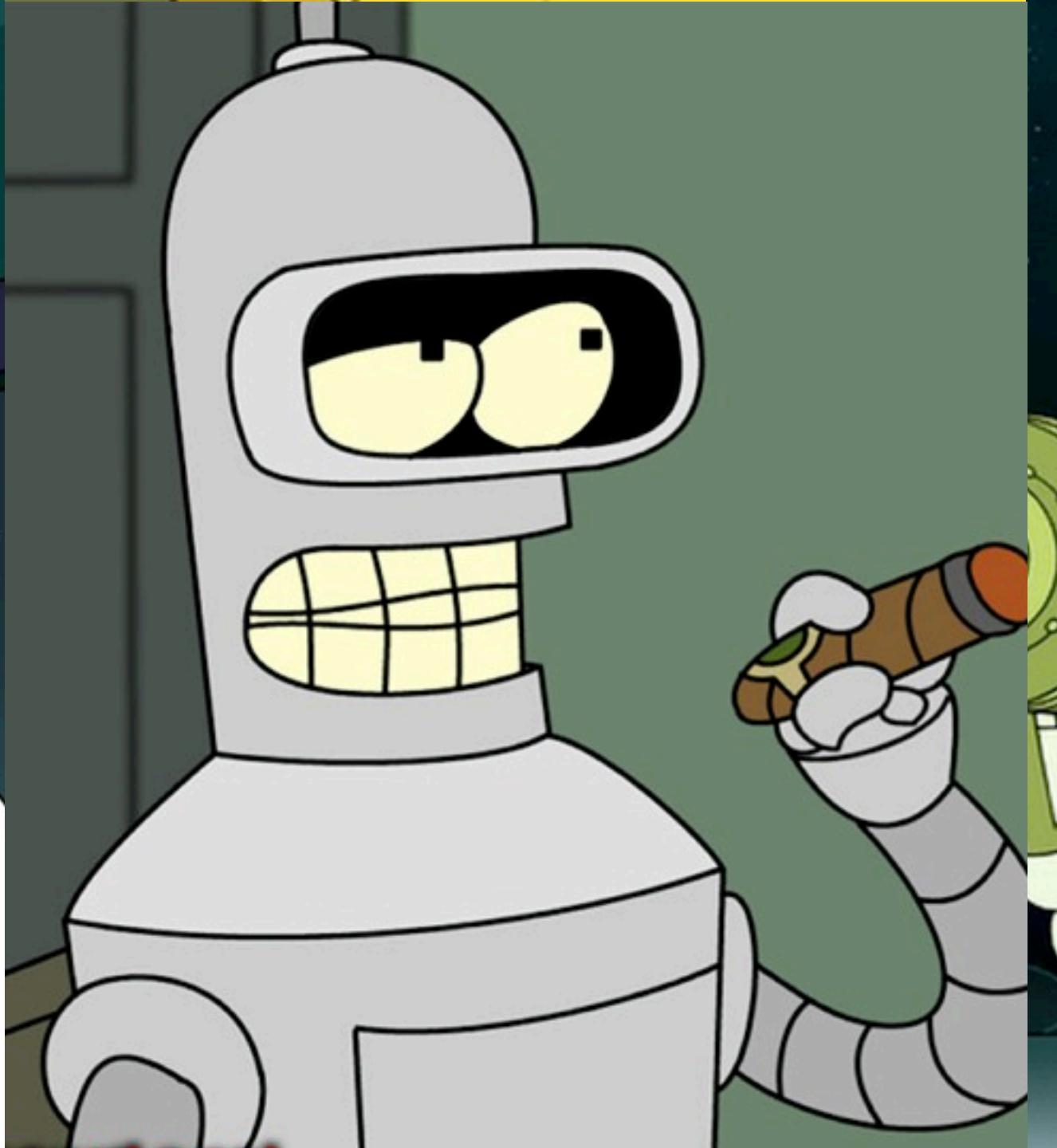
PLEASE
— TELL ME ABOUT —
YOUR RESEARCH STUDY

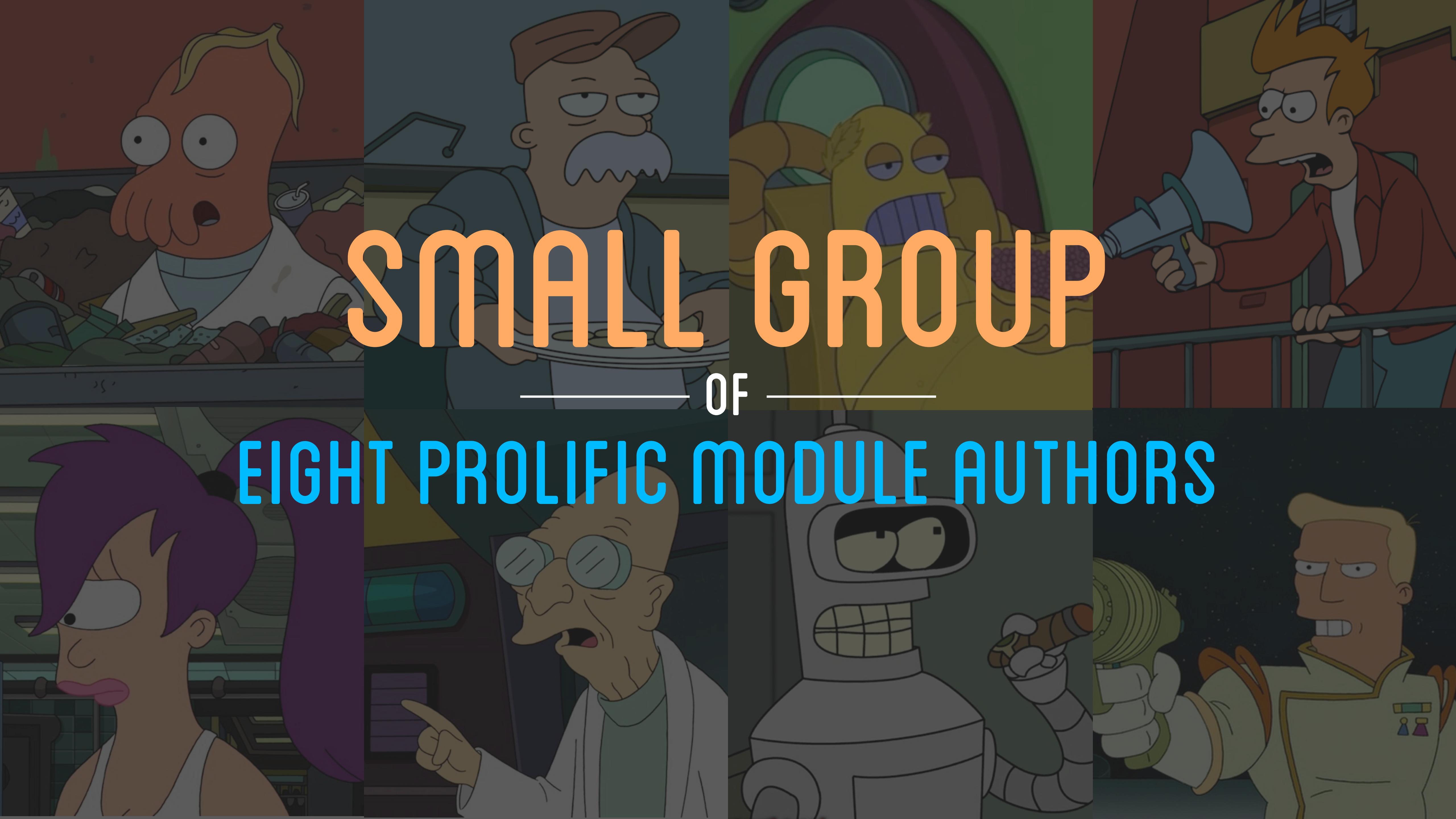
A person wearing a black mask and a green and orange patterned shawl, looking down.

— LIKE ALL RESEARCH —
WE ASKED A
— A QUESTION —

A large, diverse crowd of people is gathered in a large auditorium, sitting on the floor in rows. Many individuals are wearing lanyards with badges, suggesting a conference or event setting. The lighting is dim, with spotlights illuminating the crowd from above.

— HOW DO —
MODULE AUTHORS
— ANSWER QUESTIONS ABOUT —
THEIR COMMUNITIES?





small group
of
eight prolific module authors

— THE —

most effective

— FEEDBACK MECHANISMS —

USED MOST INFREQUENTLY

GITHUB ISSUES

WRITE A BLOG POST

TWITTER DISCUSSIONS

EMAIL A MAILING LIST

GITHUB ISSUES BUT

WRITE HOW DO WE

ANSWER

A QUESTION LIKE THIS?

EMAIL A MAILING LIST

— BY USING —



— OF COURSE —



— REPRESENTS A —
MULTI-DEMENSIONAL
GRAPH



A wide-angle, high-angle aerial photograph of the New York City skyline at sunset. The sky is filled with dramatic, orange and yellow-hued clouds. In the foreground, the dense cluster of Manhattan skyscrapers is visible, with the Empire State Building standing prominently on the left. To the right, the Hudson River and the New Jersey shore are visible. The overall atmosphere is one of a bustling, iconic urban environment.

— I AM —
FROM NEW YORK



— I AM —
FROM NEW YORK
— AND SO IS —







— WAS ALWAYS —
ABOUT

CoRoEAM

Dolla Dolla Bill Y'all

A fluffy, light brown dog with dark brown ears and a white patch on its chest is wearing a pair of dark-rimmed glasses. It is sitting at a laptop, with its front paws resting on the keyboard. The dog is looking directly at the camera with a slightly curious expression. A blue and white checkered collar is around its neck. The background is a plain, light-colored wall.

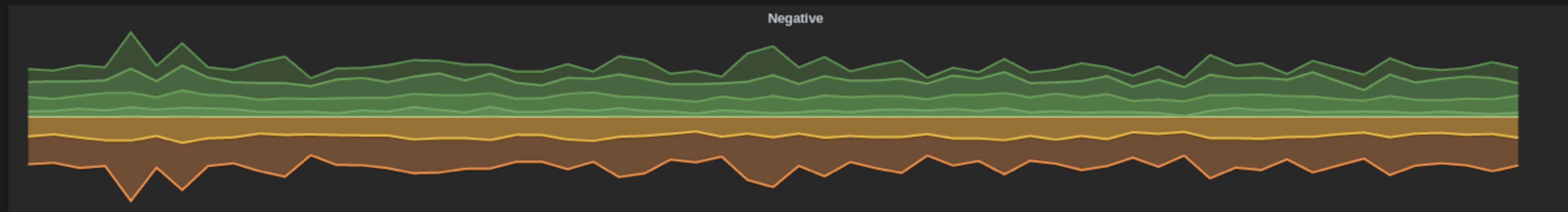
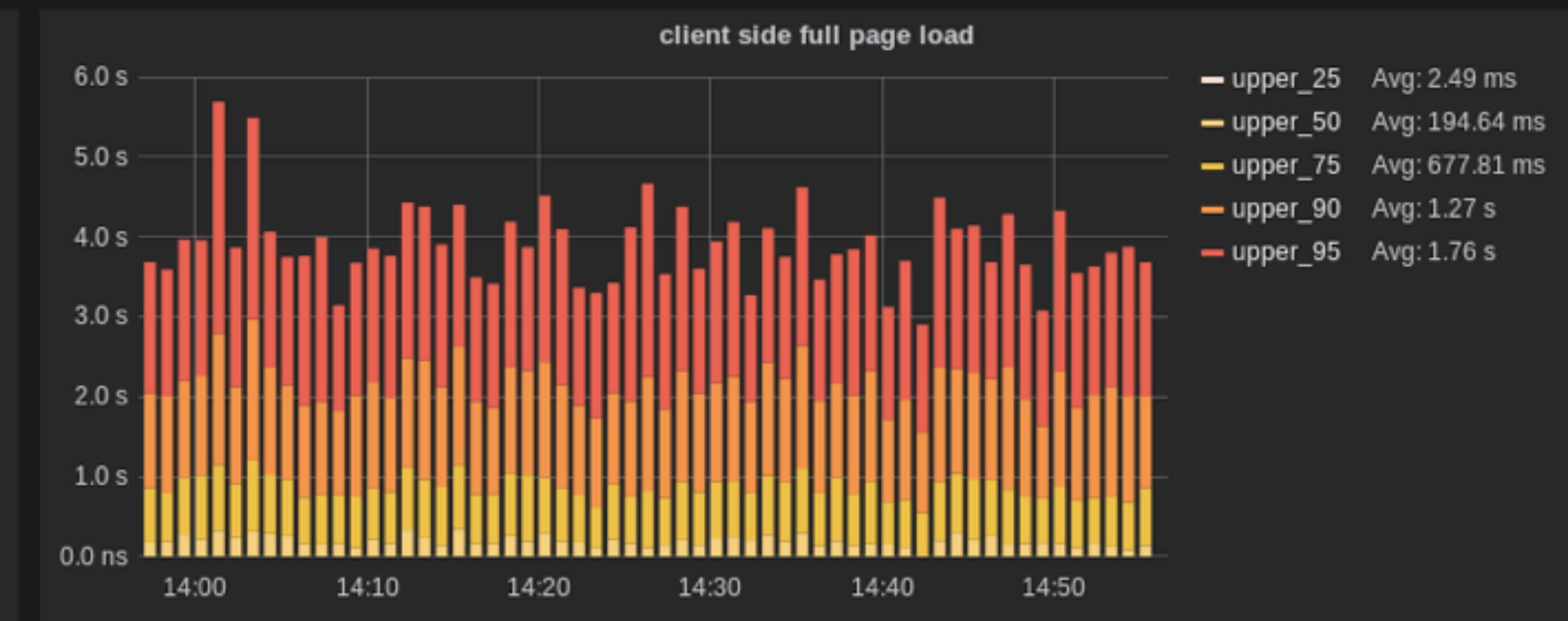
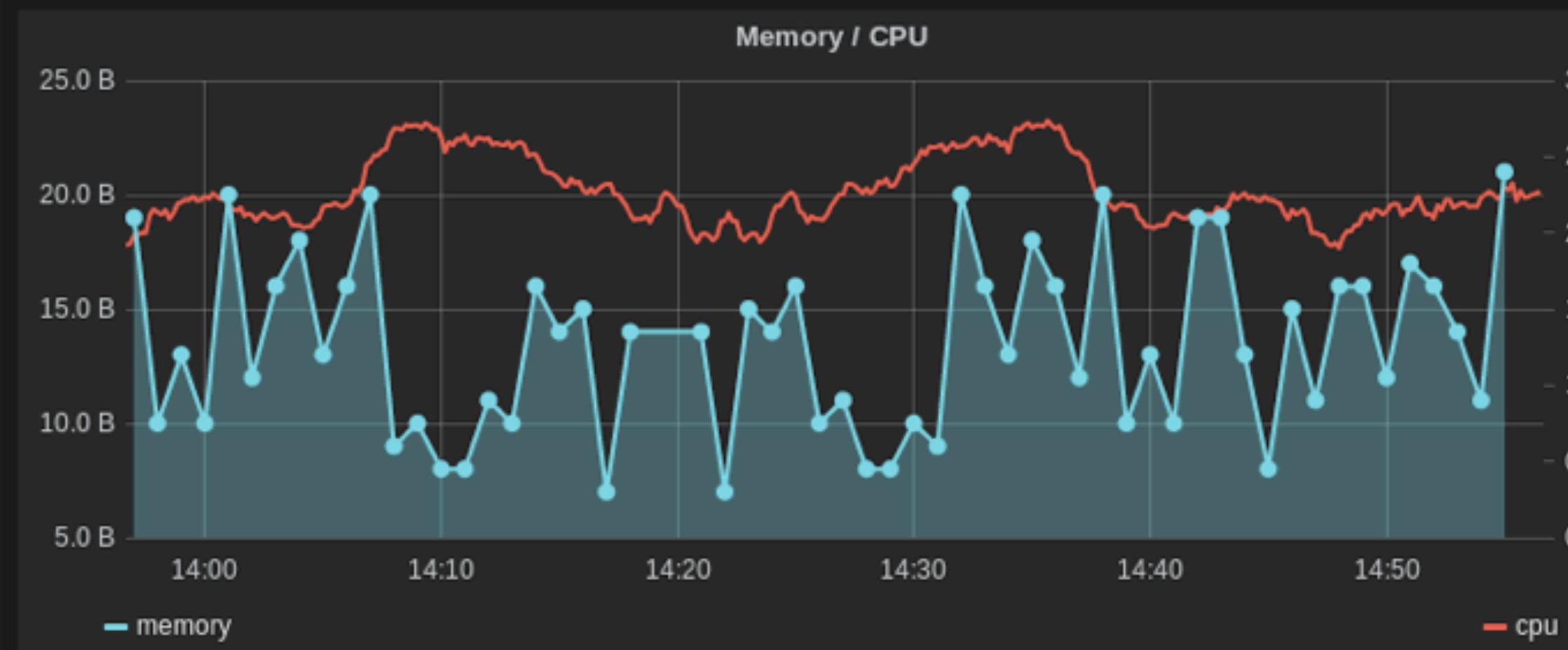
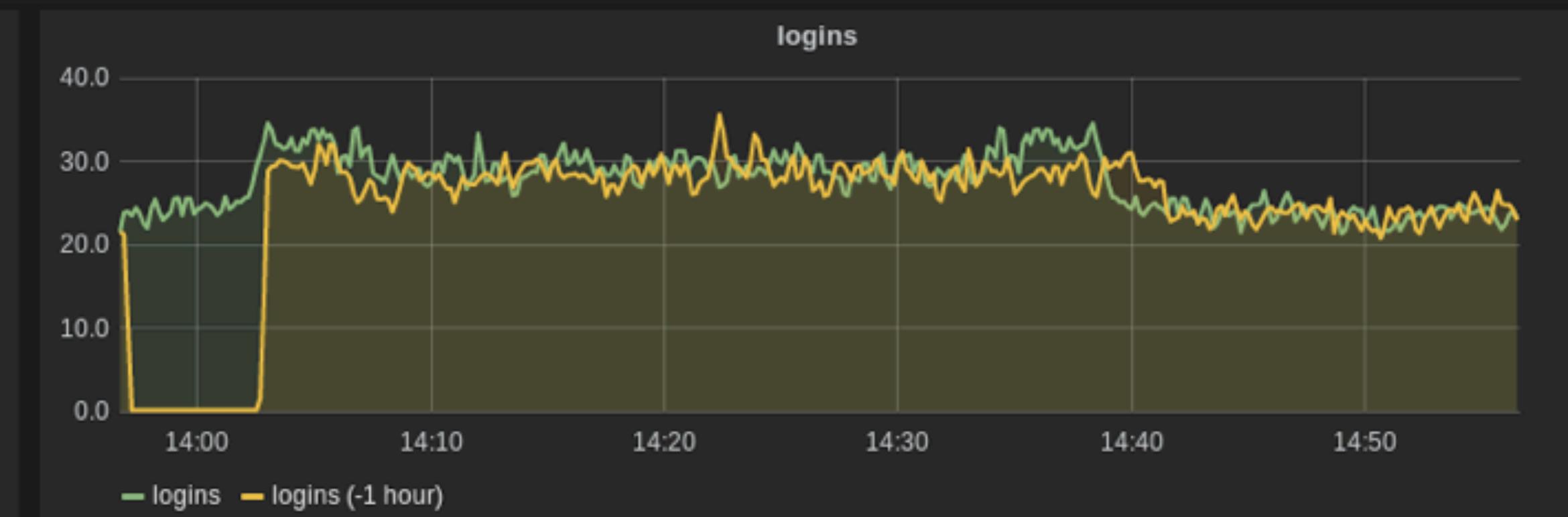
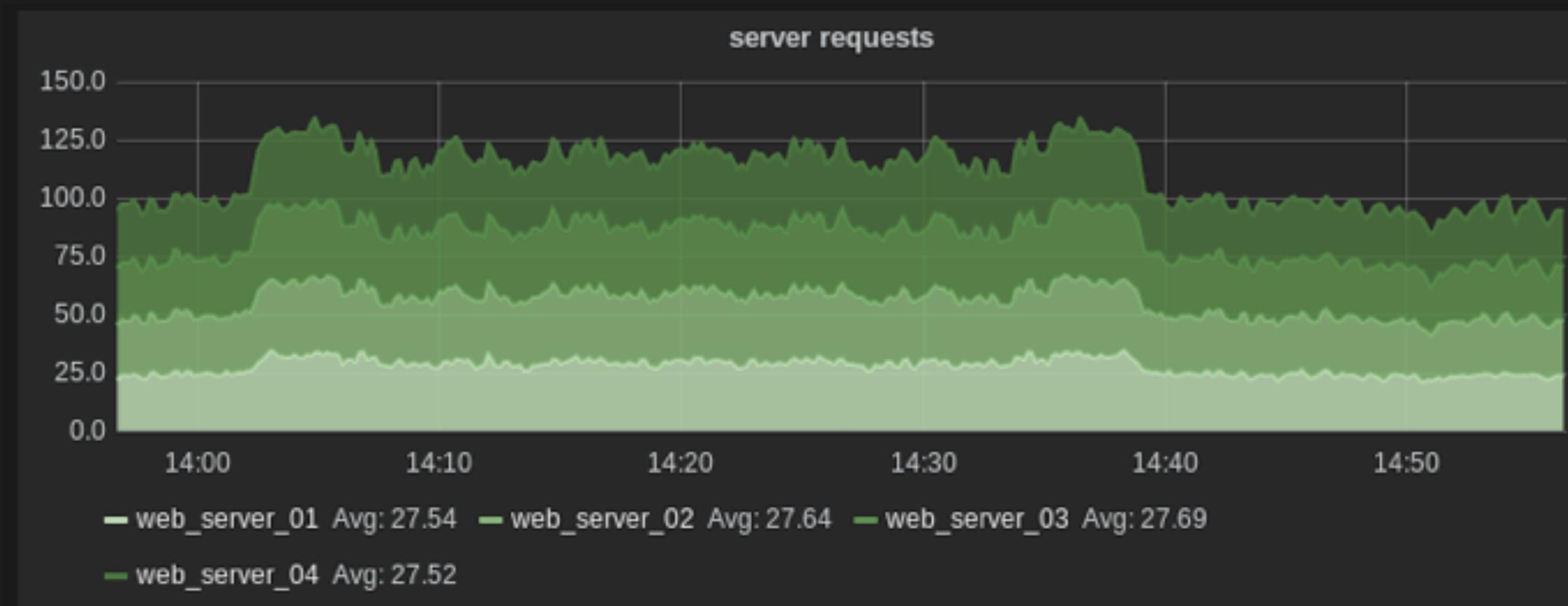
— BUT I AM ALSO —
AN ENGINEER

G. R. E. A. M.

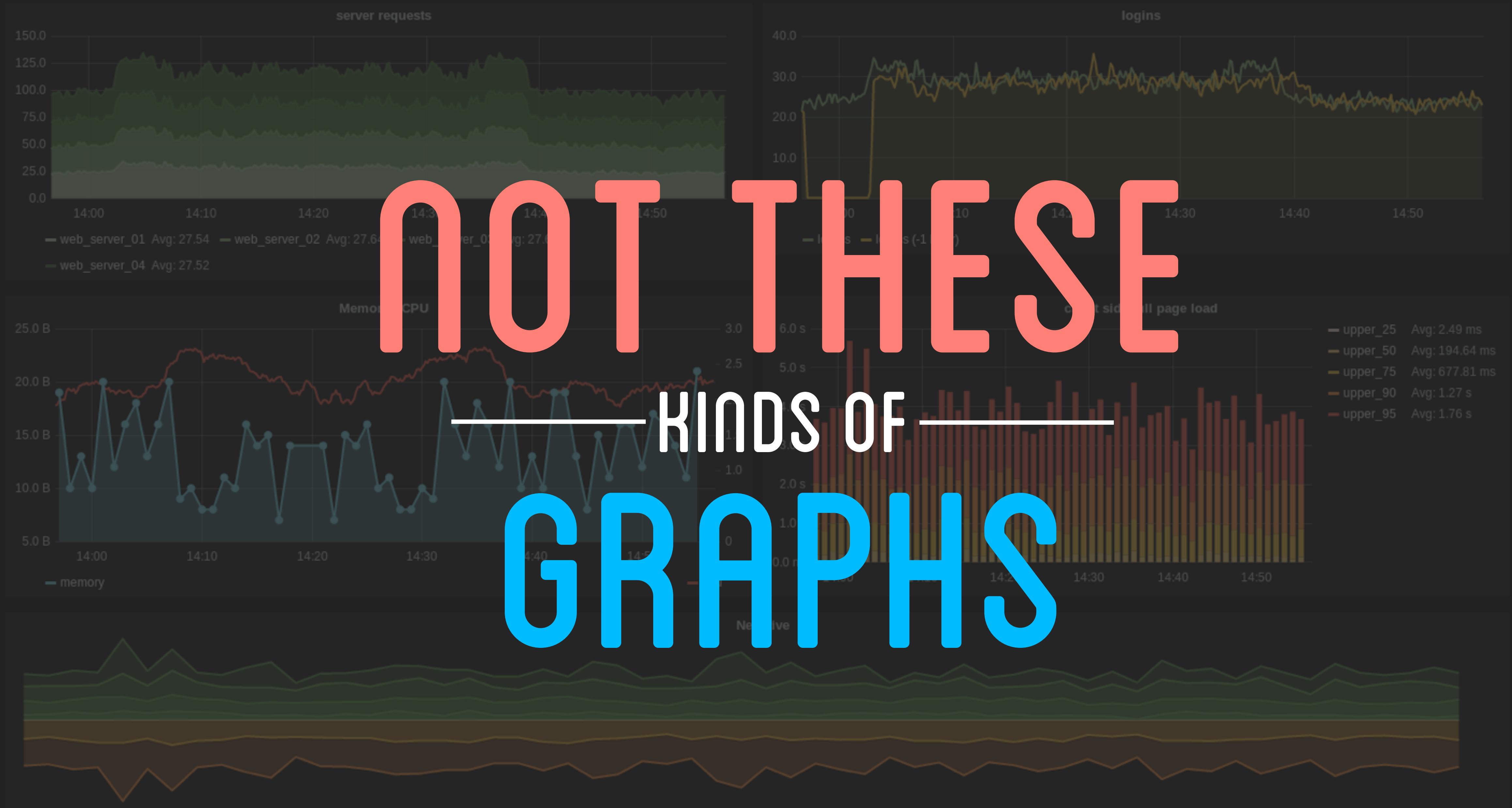
GRAPHS
RULE
EVERYTHING
AROUND
ME

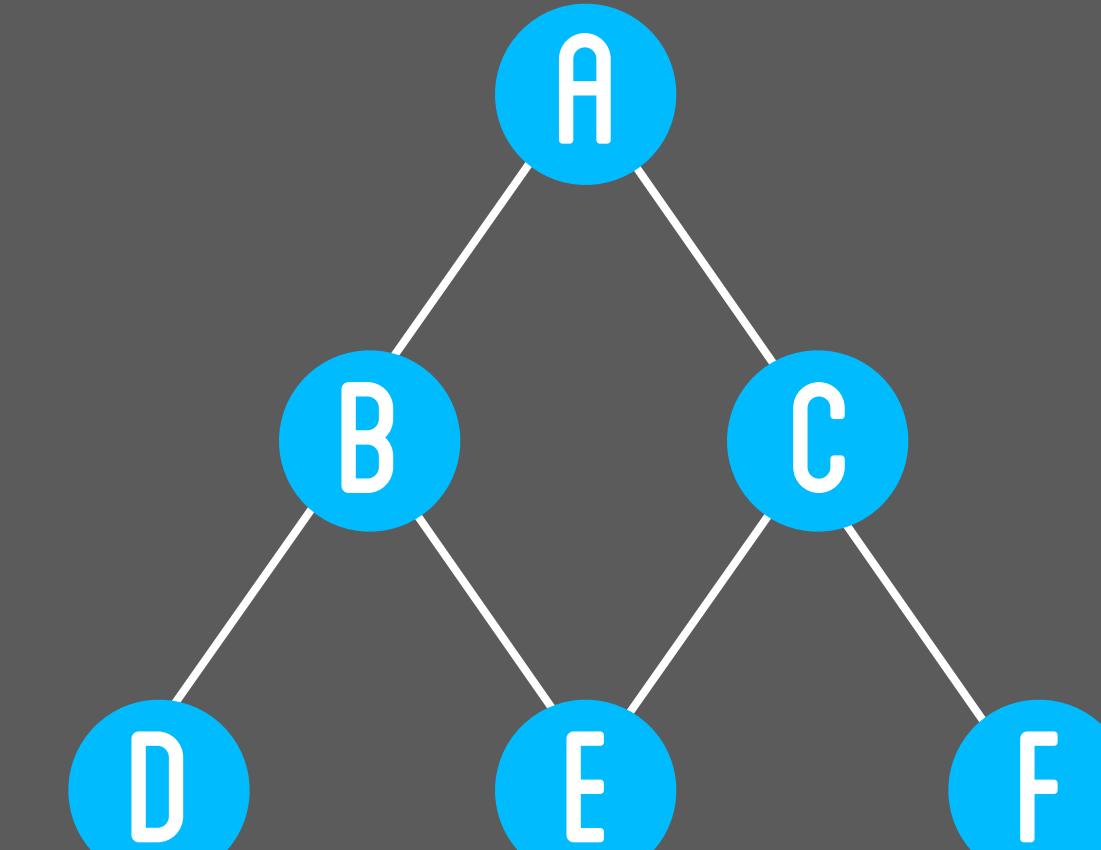
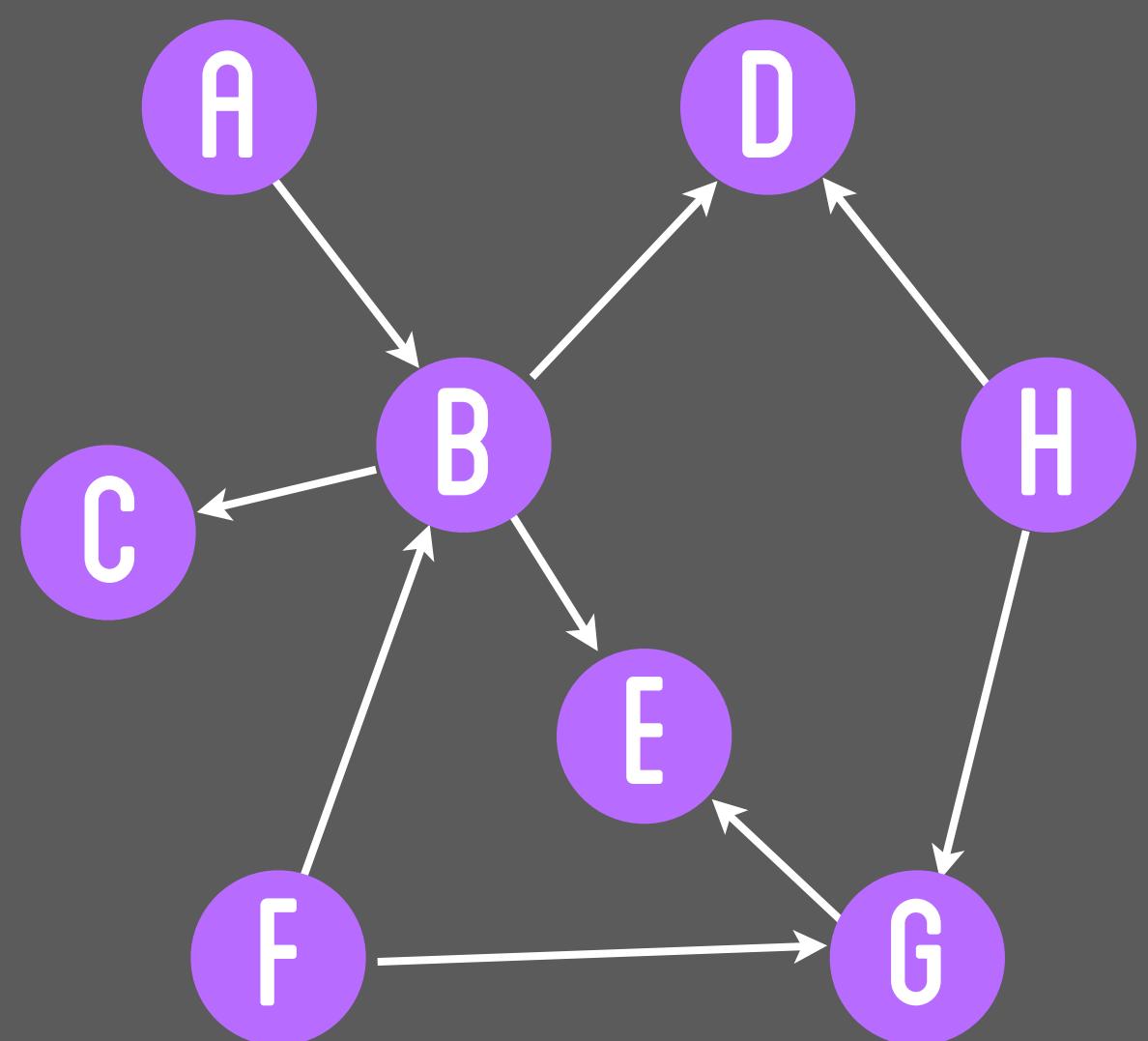
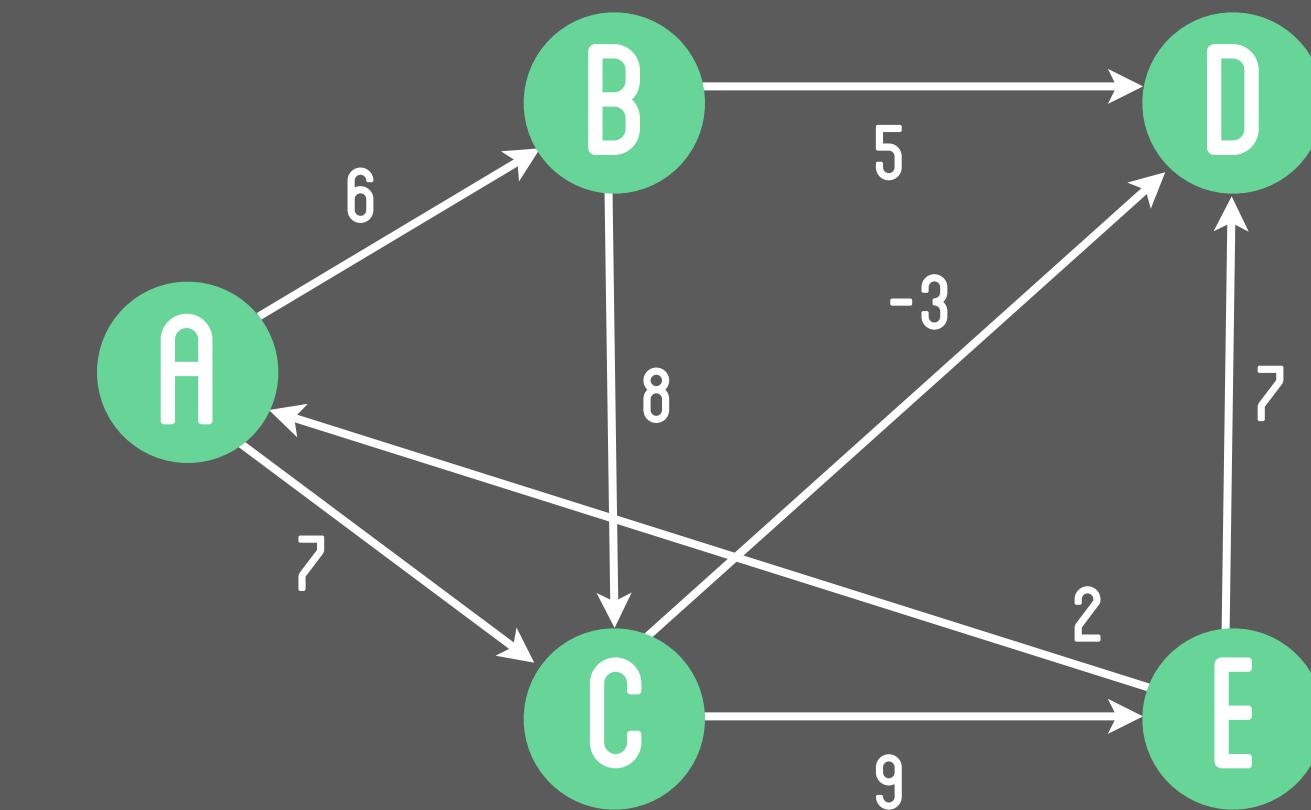
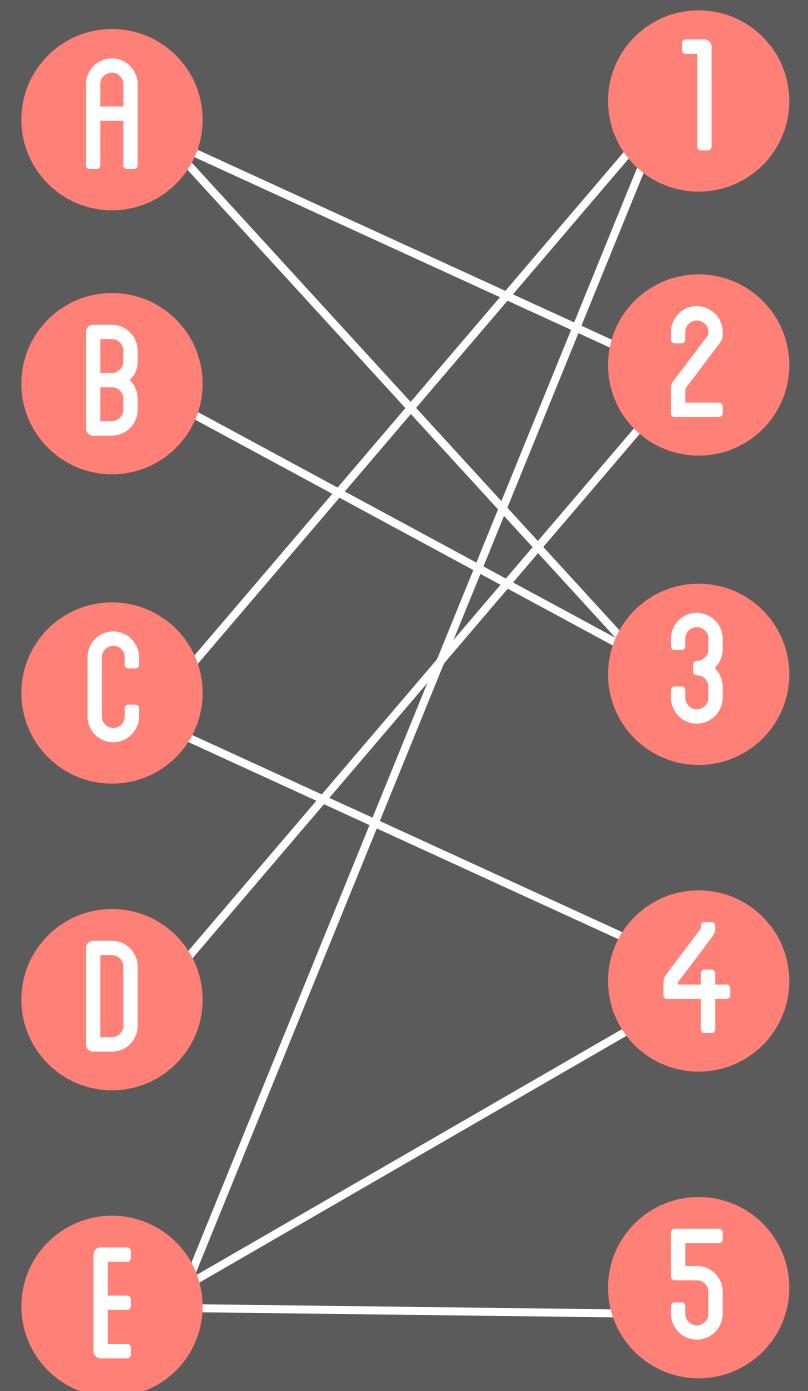
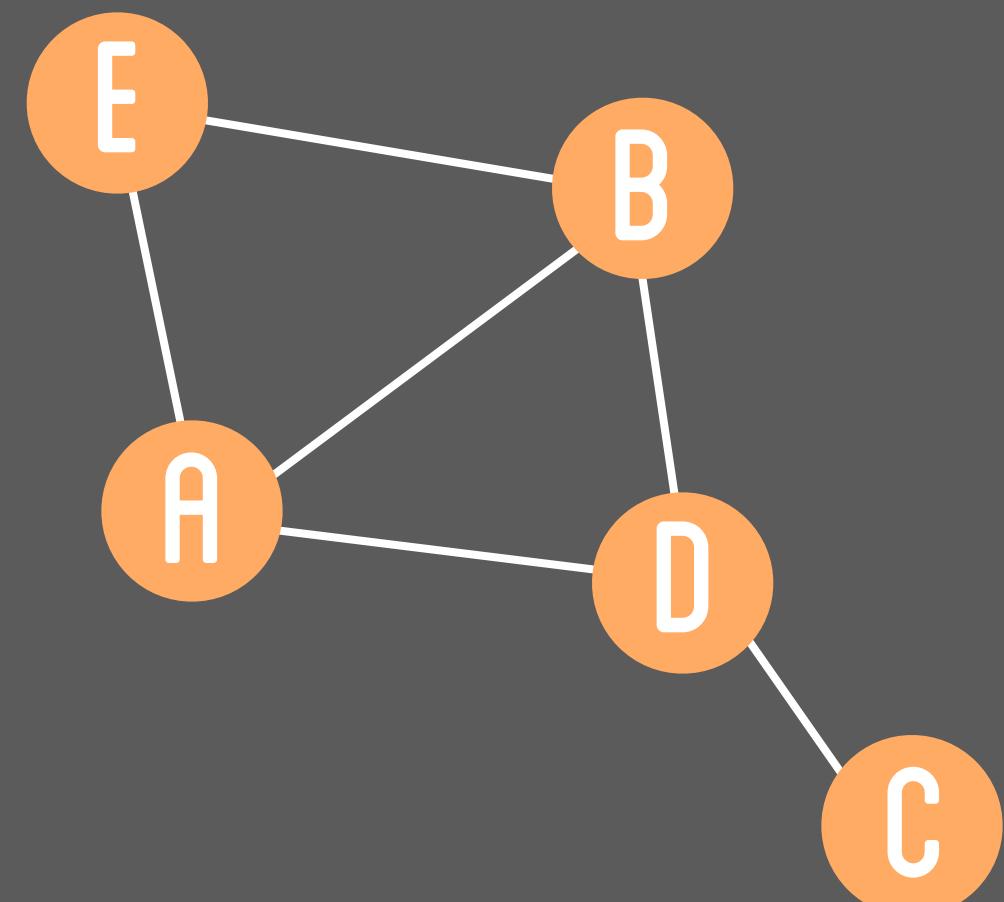
GRAPHS
RULE
EVERYTHING
AROUND
ME

DOLLA DOLLA BILL Y'ALL

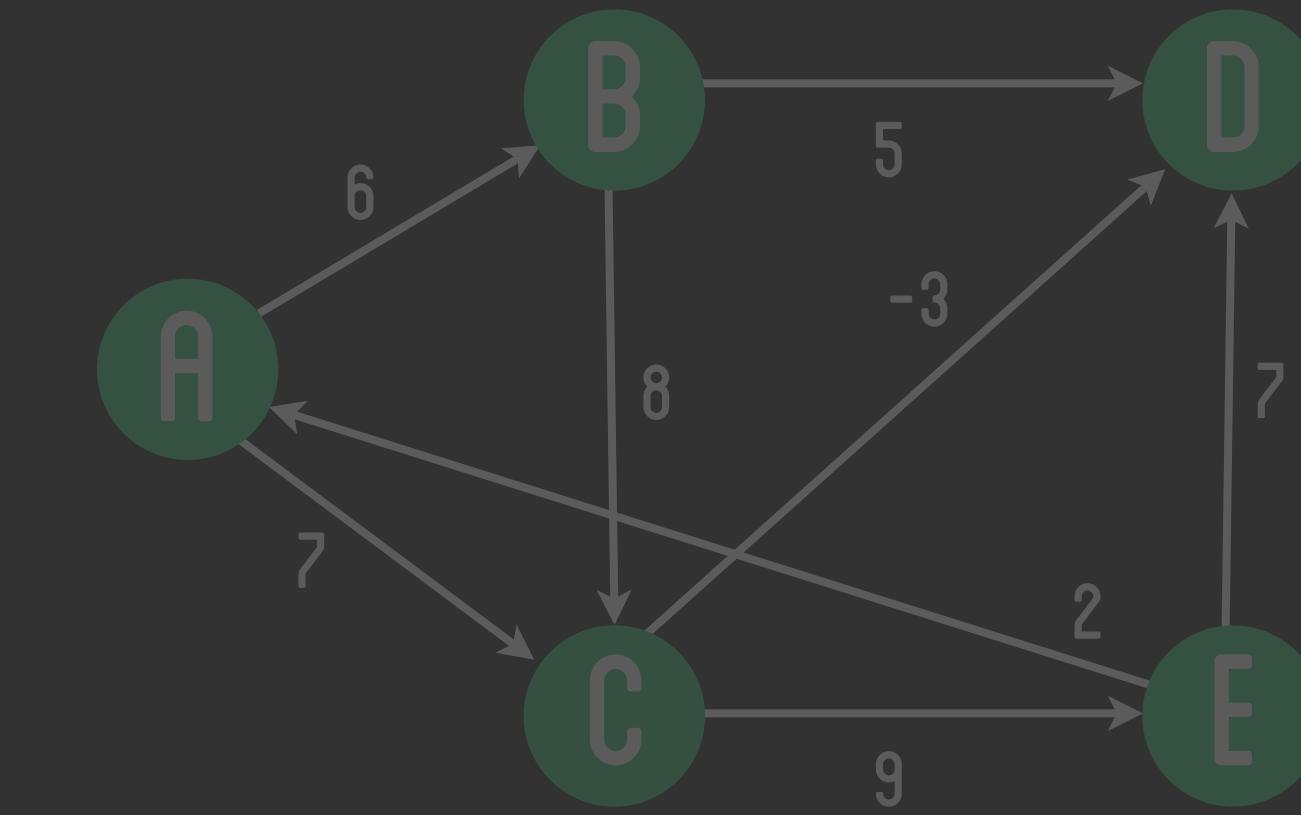
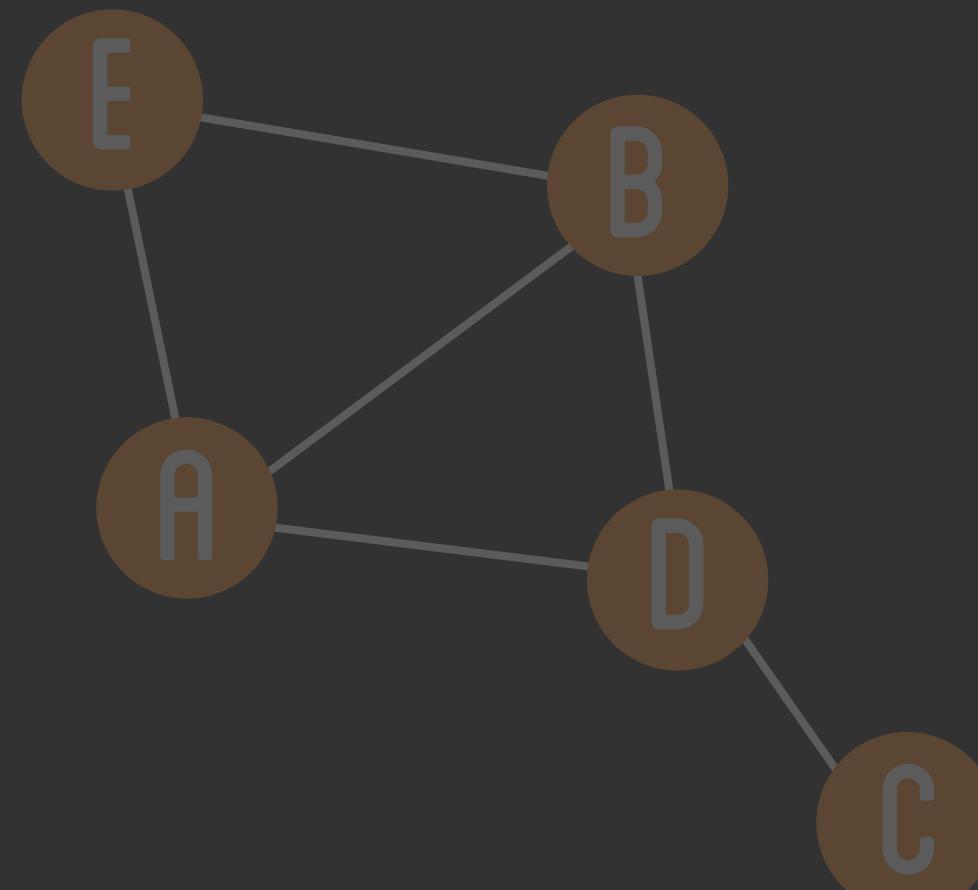
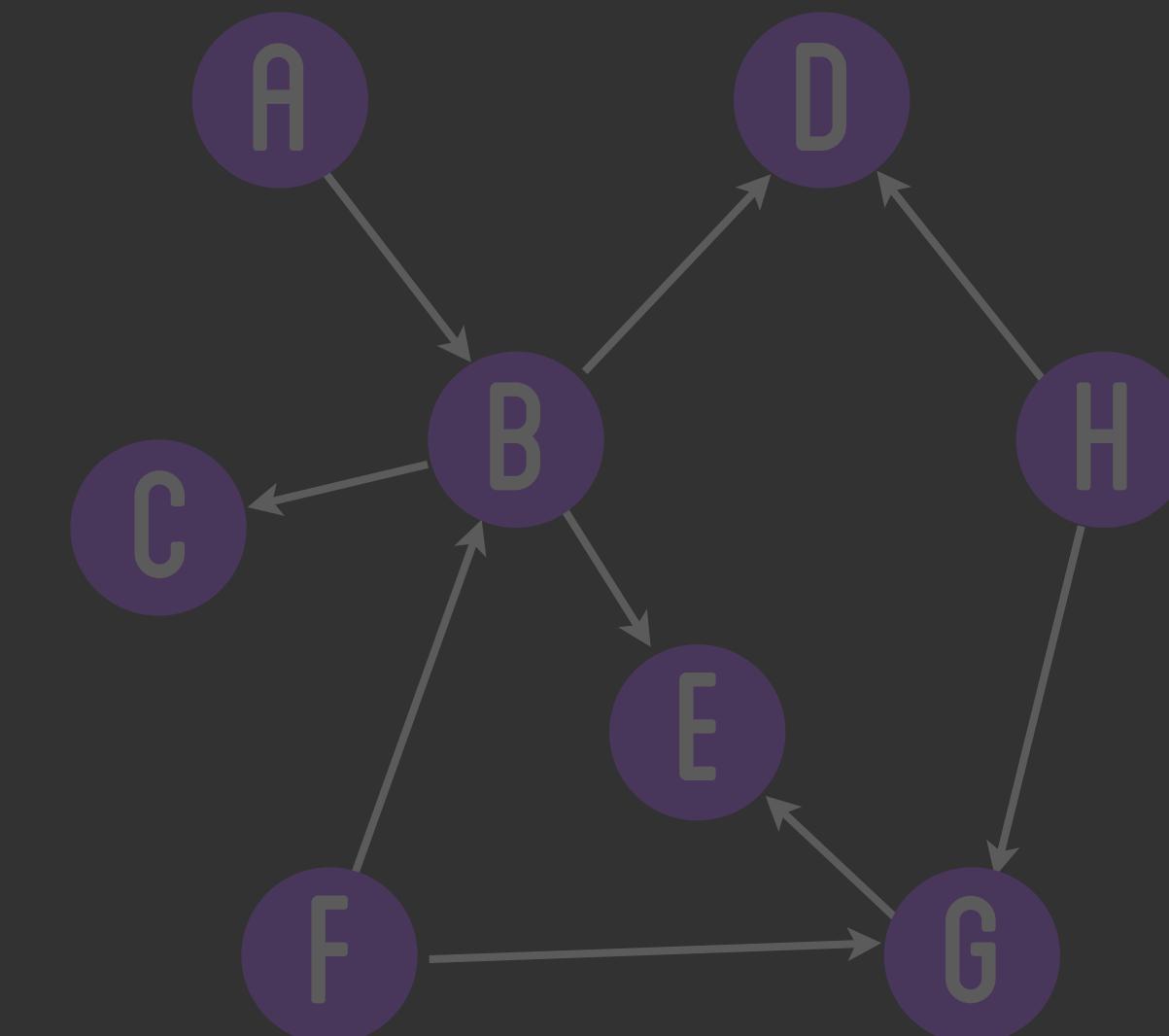


NOT THESE
KINDS OF —
GRAPHS

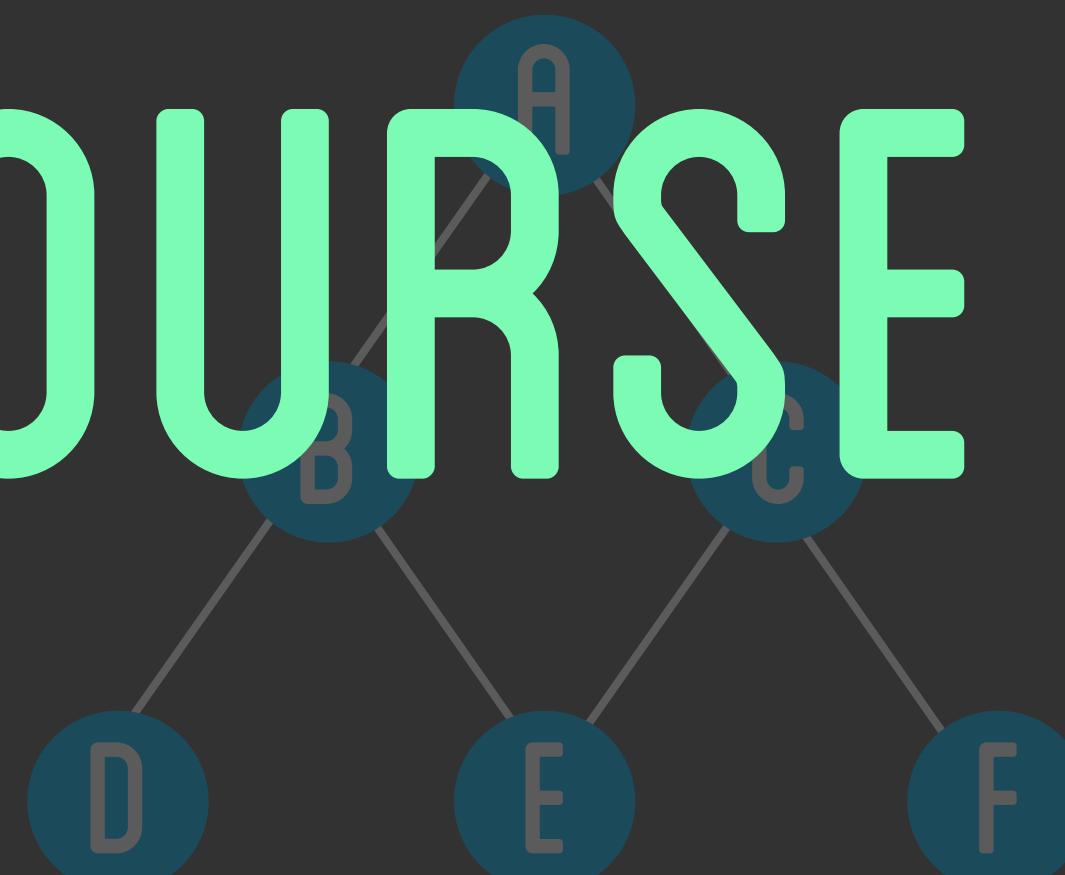
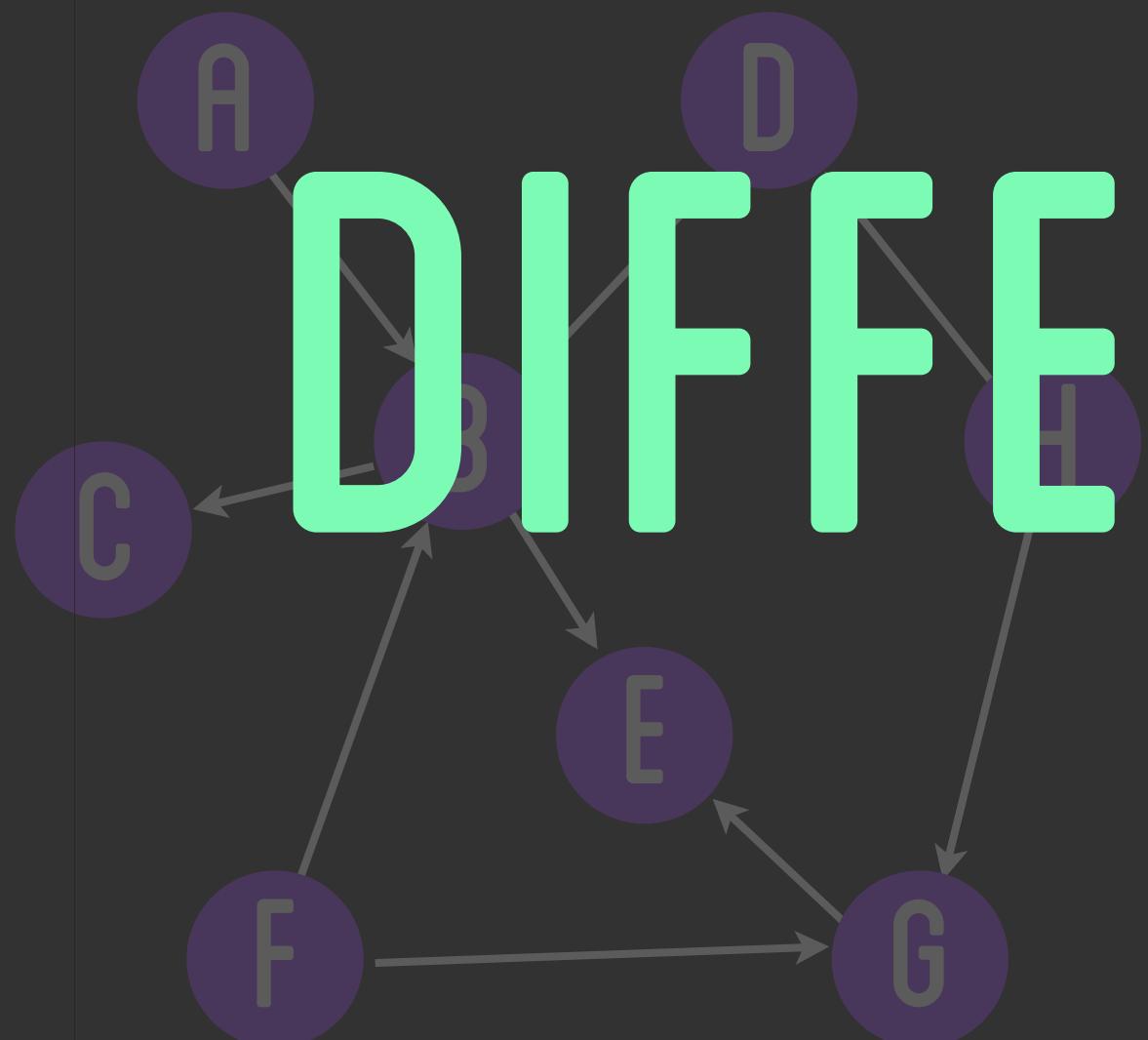
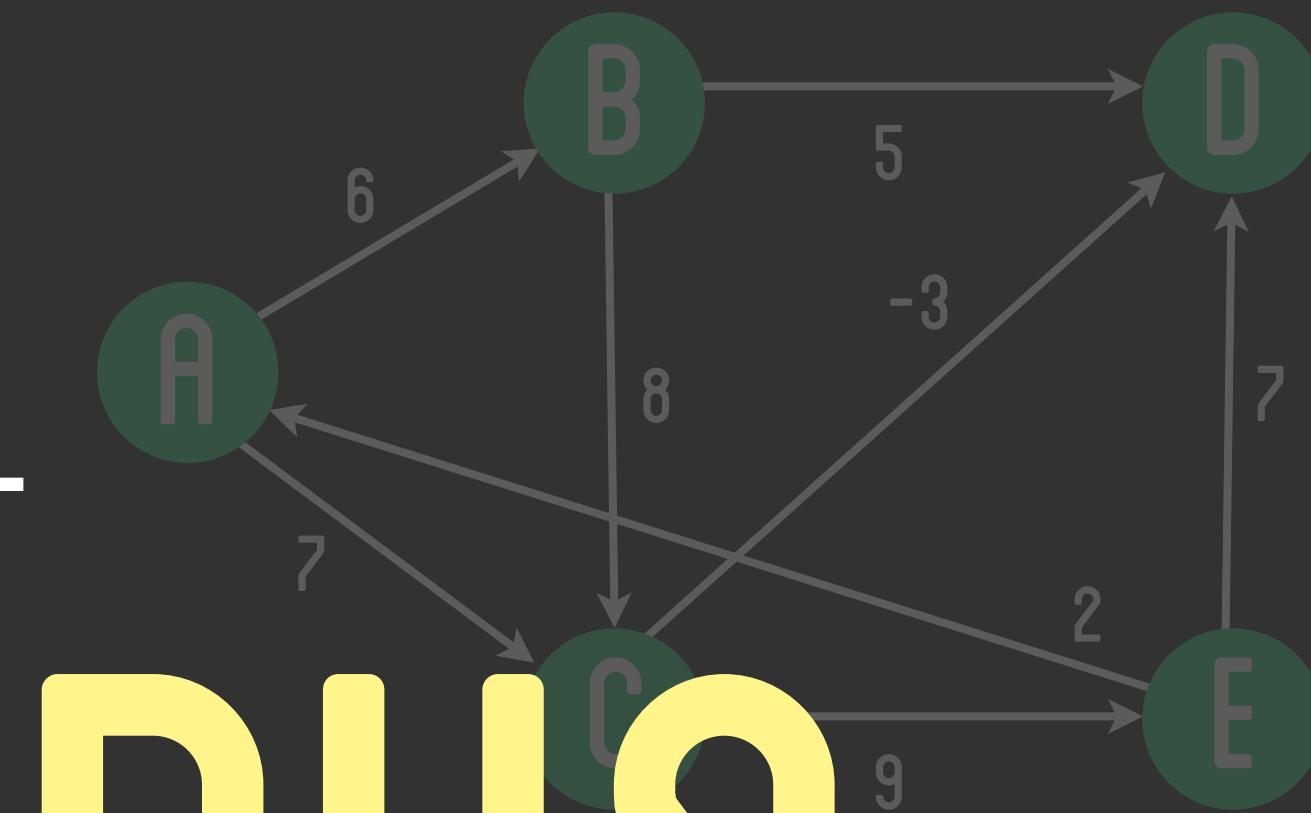
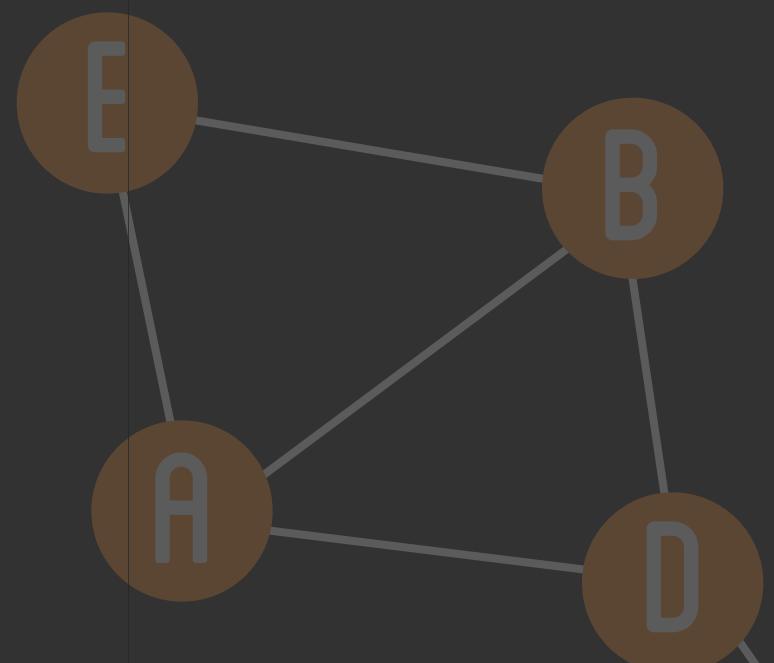


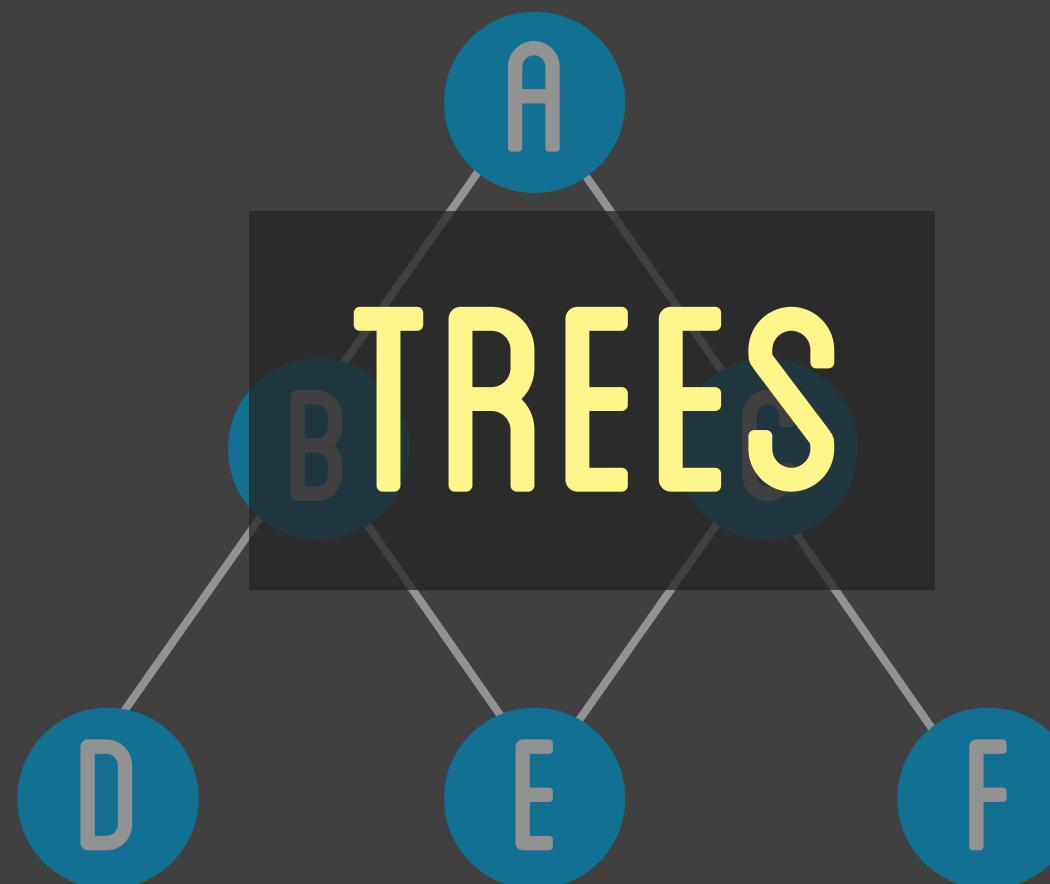
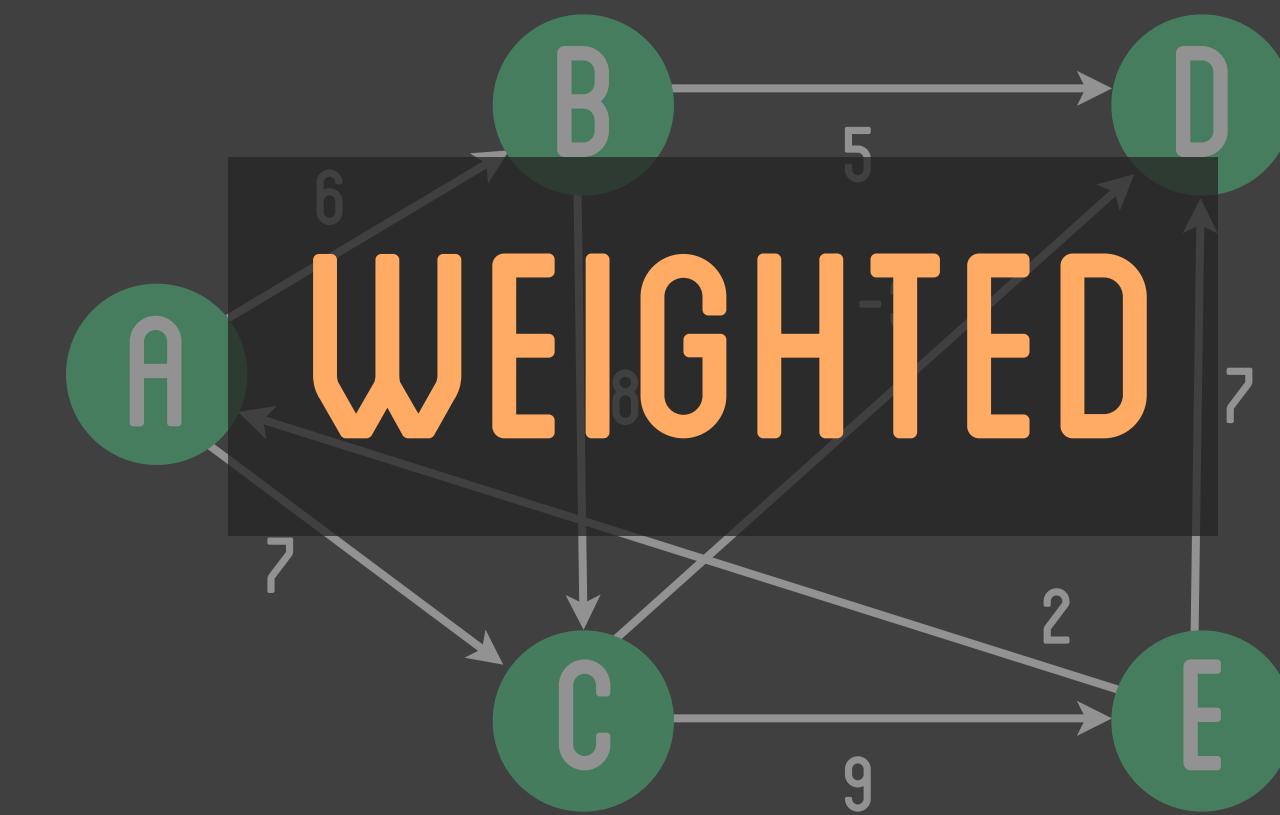
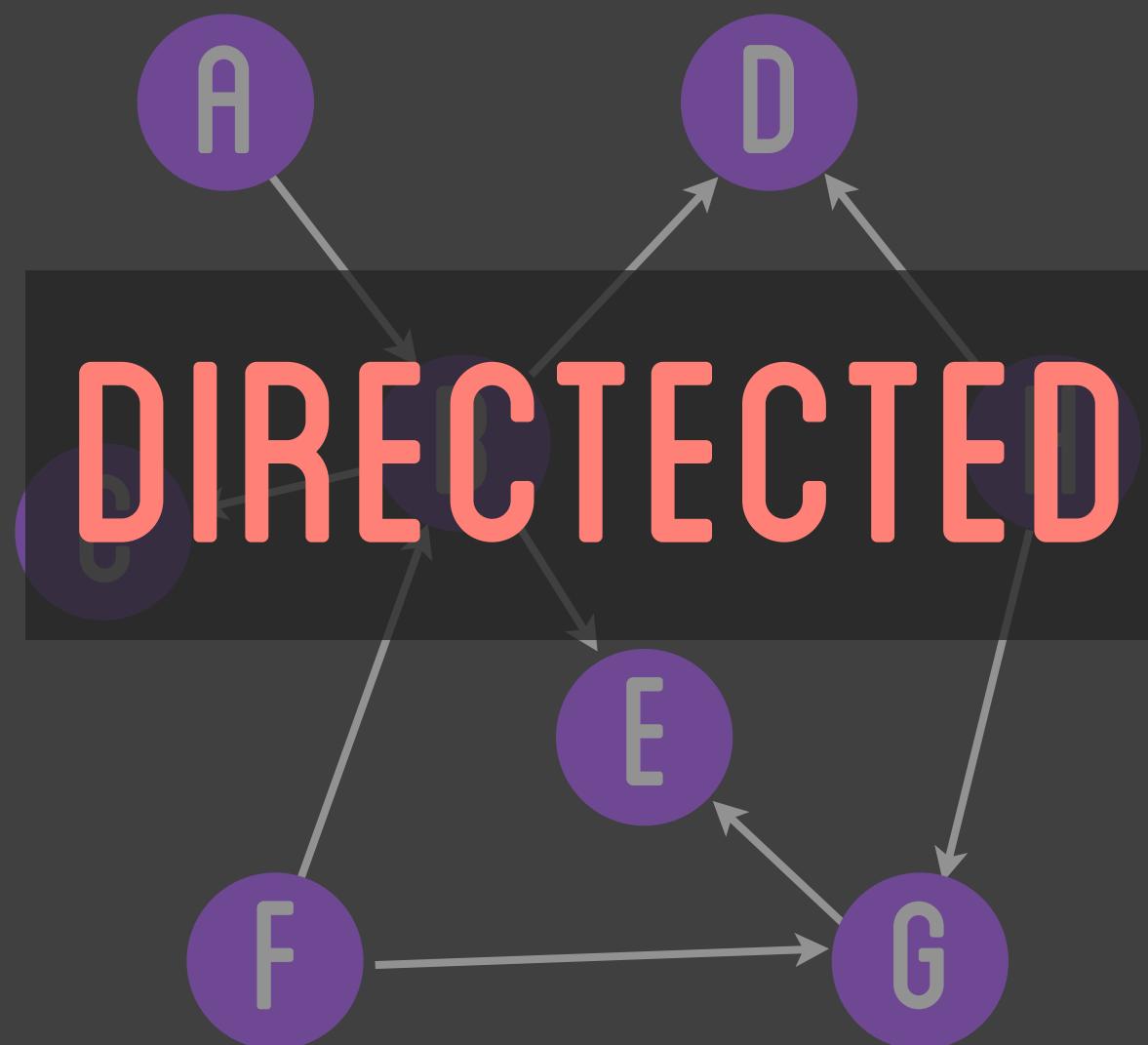
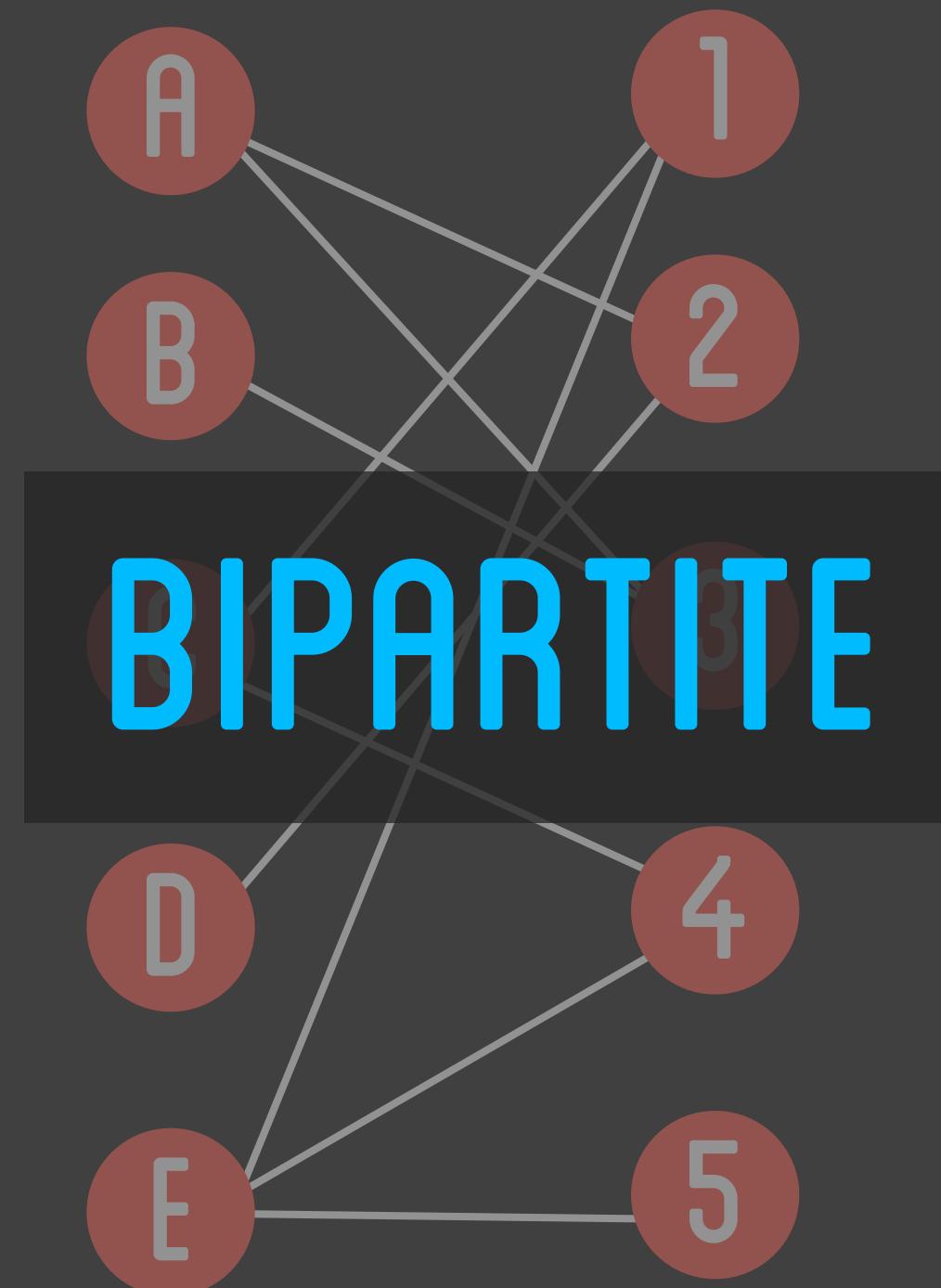
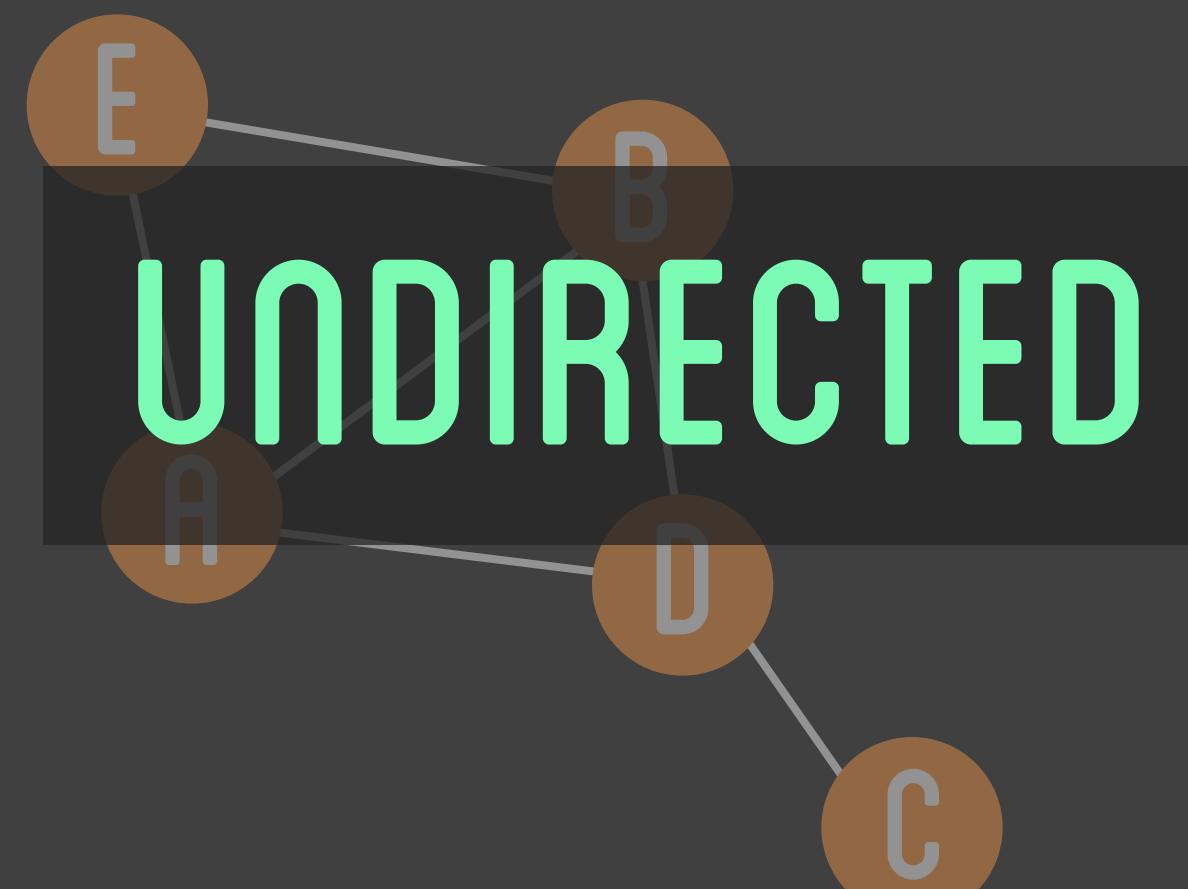


THESE KINDS OF GRAPHS



ALL OF
THESE GRAPHS
ARE
DIFFERENT, OF COURSE





TRANSPORTATION NETWORKS

INFORMATION NETWORKS

MOLECULAR CHEMISTRY

WIRELESS NETWORKS

MAJOR LEAGUE BASEBALL

DEPENDENCY MANAGEMENT

TRANSPORTATION NETWORKS

INDEPENDENCY GRAPHS?

MOLECULAR CHEMISTRY
JE NE SAIS PAS —

WIRES SOUNDS COMPLICATED FOR

MAJOR 10AM DANS LE MAINTAIN

DEPENDENCY MANAGEMENT

— HIPSTER CAT SAYS —



— HIPSTER CAT SAYS —
“ACTUALLY, IT’S NOT SO BAD”



WITH A PACKAGE.JSON FILE

```
{  
  "name": "pkg-a",  
  "dependencies": {  
    "pkg-b": "~1.0.4",  
    "pkg-c": "~2.1.3"  
  },  
  "devDependencies": {  
    "pkg-d": "~3.1.2"  
  },  
  "main": "./index.js"  
}
```

WITH A PACKAGE.JSON FILE

```
{  
  "name": "pkg-a",  
  "dependencies": {  
    "pkg-b": "~1.0.4",  
    "pkg-c": "~2.1.3"  
  },  
  "devDependencies": {  
    "pkg-d": "~3.1.2"  
  },  
  "main": "./index.js"  
}
```



n_a

WITH A PACKAGE.JSON FILE

```
{  
  "name": "pkg-a",  
  "dependencies": {  
    "pkg-b": "~1.0.4",  
    "pkg-c": "~2.1.3"  
  },  
  "devDependencies": {  
    "pkg-d": "~3.1.2"  
  },  
  "main": "./index.js"  
}
```



n_a

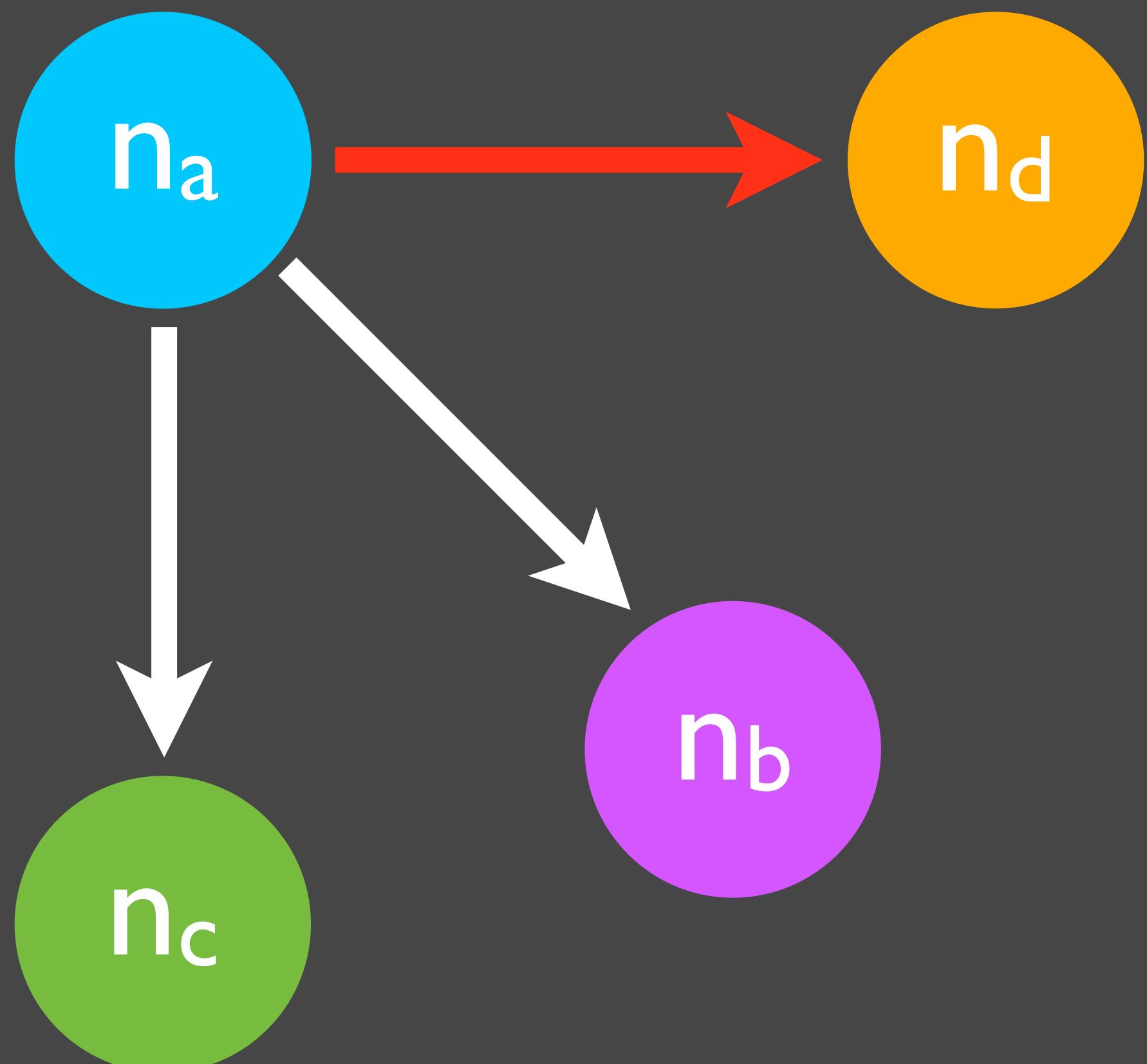
n_b

n_c

n_d

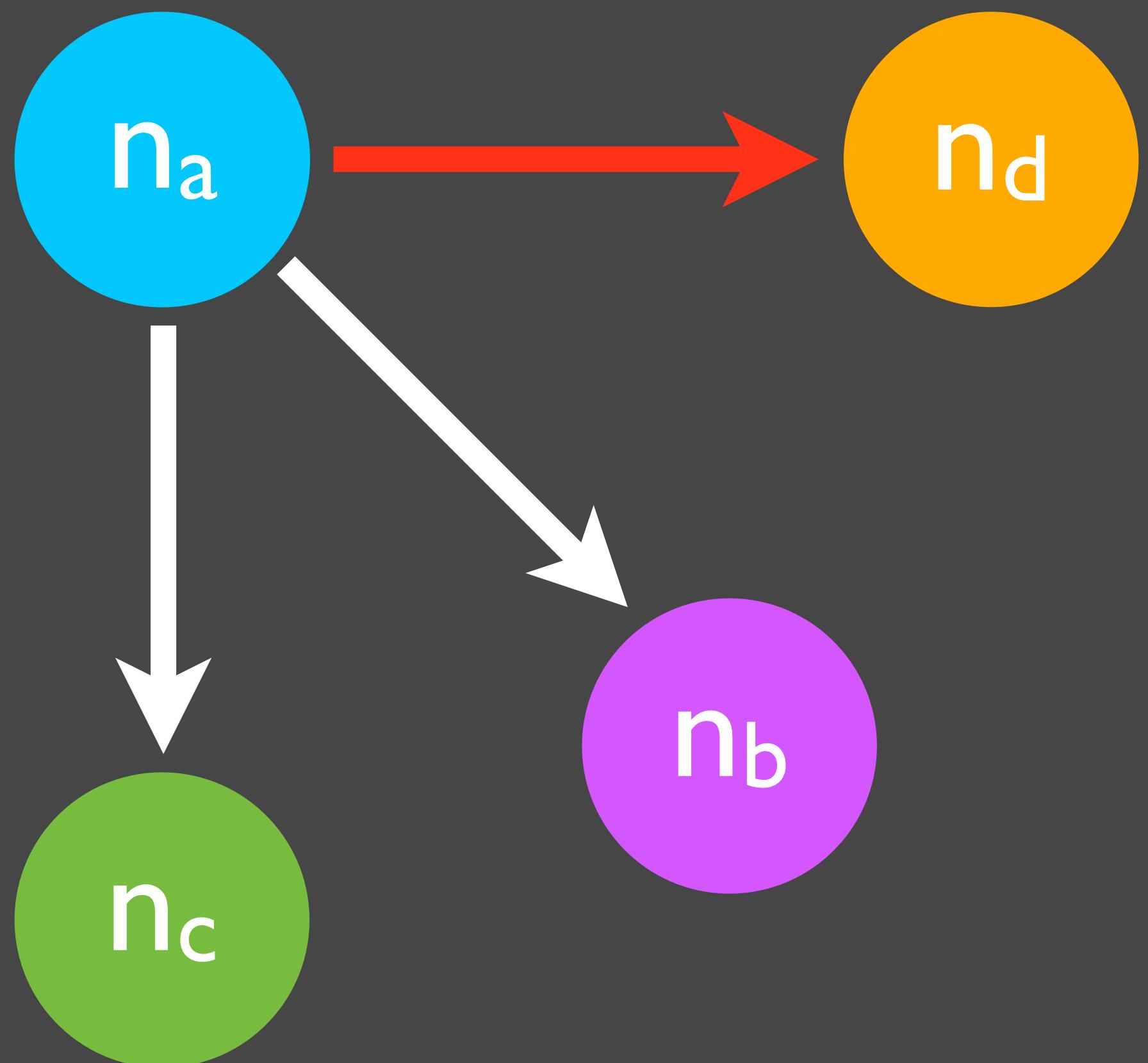
WITH A PACKAGE.JSON FILE

```
{  
  "name": "pkg-a",  
  "dependencies": {  
    "pkg-b": "~1.0.4",  
    "pkg-c": "~2.1.3"  
  },  
  "devDependencies": {  
    "pkg-d": "~3.1.2"  
  },  
  "main": "./index.js"  
}
```



WITH A PACKAGE.JSON FILE

```
{  
  "name": "pkg-a",  
  "dependencies": {  
    "pkg-b": "~1.0.4",  
    "pkg-c": "~2.1.3"  
  },  
  "devDependencies": {  
    "pkg-d": "~3.1.2"  
  },  
  "main": "./index.js"  
}
```



Now imagine this for 100,000+ packages!



CASCADIAJS 2014

JULY 31ST & AUGUST 1ST → PORTLAND, OR

LXJS
2014

THESE TALKS

— ARE ALL ALREADY —
ONLINE WITH VIDEOS



CASCADIAJS 2014

JULY 31ST & AUGUST 1ST → PORTLAND, OR



DR. EMMETT OCTOCAT SAYS
“YOU’VE GOT TO COME BACK WITH ME.
BACK TO GITHUB!”

INDEXZERO/NPM-CODEPENDENCIES

INDEXZERO/NPM-COMP-STAT-WWW

INDEXZERO/NPM-STATIC-STATS

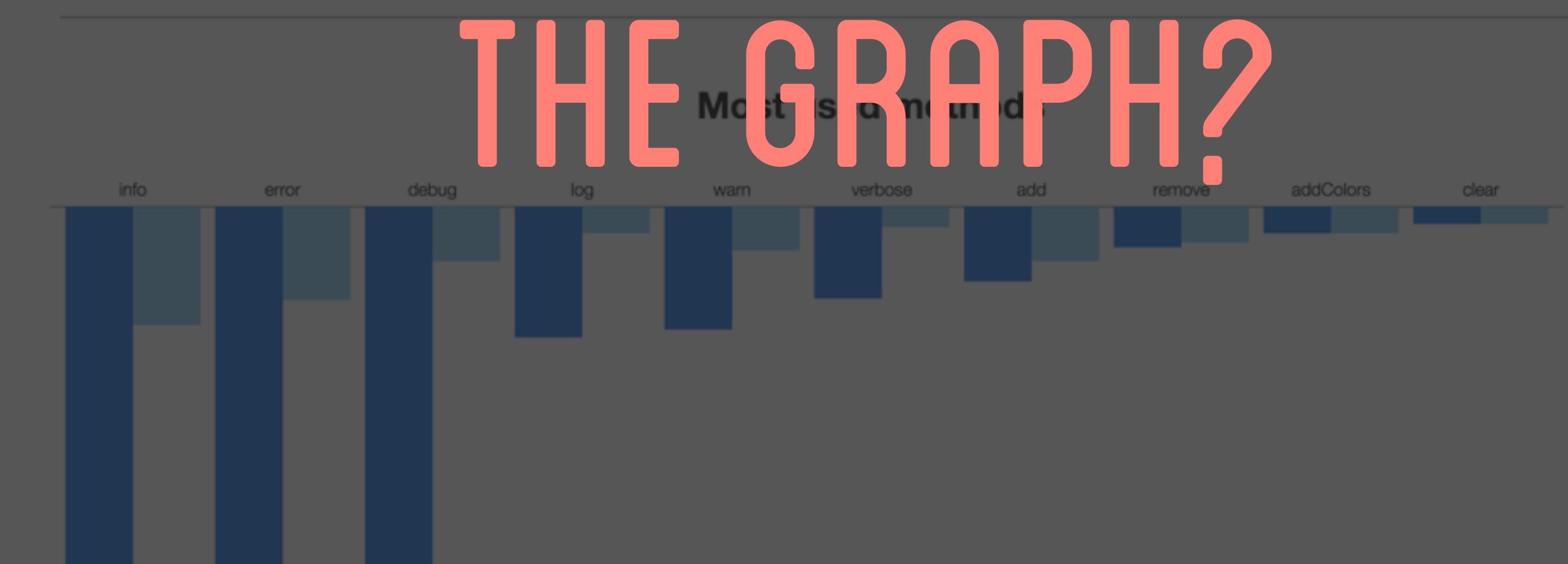
INDEXZERO/NPM-PIPELINE

DR. EMMETT OCTOCAT SAYS

“YOU’VE GOT TO COME BACK WITH ME.
BACK TO GITHUB!”

Codependency Graphs

WHAT OTHER
QUESTIONS
CAN WE ASK FROM
THE GRAPH?



```
npm http 304 https://us.registry.nodejitsu.com/stack-trace
npm http 304 https://us.registry.nodejitsu.com/eventemitter2
npm http 304 https://us.registry.nodejitsu.com/pkginfo
npm http 304 https://us.registry.nodejitsu.com/read
npm info attempt registry request try #1 at 18:49:39
npm http request GET https://us.registry.nodejitsu.com/mute-stream
npm http 304 https://us.registry.nodejitsu.com/revalidator
npm http 304 https://us.registry.nodejitsu.com/lru-cache
npm http 304 https://us.registry.nodejitsu.com/minimist
npm http 304 https://us.registry.nodejitsu.com/event-stream
npm http 304 https://us.registry.nodejitsu.com/mute-stream
npm http 304 https://us.registry.nodejitsu.com/sigmund
```

<u>Package</u>	<u>Current</u>	<u>Wanted</u>	<u>Latest</u>	<u>Location</u>
colors	0.6.2	0.6.2	1.0.3	colors
nssocket	0.5.1	0.5.2	0.5.2	nssocket
vows	0.7.0	0.7.0	0.8.0	vows
request	2.47.0	2.48.0	2.48.0	request
colors	0.6.2	0.6.2	1.0.3	cliff > colors
optimist	0.6.0	0.6.0	0.6.1	nconf > optimist
async	0.2.9	0.2.9	0.9.0	nconf > async
minimist	0.0.10	0.0.10	1.1.0	optimist > minimist
async	0.2.10	0.2.10	0.9.0	utile > async
minimist	0.0.8	0.0.8	1.1.0	utile > mkdirp > minimist
ncp	0.4.2	0.4.2	1.0.1	utile > ncp
colors	0.6.2	0.6.2	1.0.3	winston > colors
async	0.2.10	0.2.10	0.9.0	winston > async
winston	0.8.0	0.8.0	0.8.3	broadway > winston
optimist	0.6.0	0.6.0	0.6.1	flatiron > optimist
revalidator	0.1.2	0.1.2	0.2.0	flatiron > prompt > revalidator

— YOU PROBABLY ASK —

THESE QUESTIONS

EVERYDAY —

WHETHER YOU KNOW IT OR NOT

```
npm http 304 https://us.registry.nodejitsu.com/stack-trace
npm http 304 https://us.registry.nodejitsu.com/eventemitter2
npm http 304 https://us.registry.nodejitsu.com/pkginfo
npm http 304 https://us.registry.nodejitsu.com/read
npm info attempt registry request try #1 at 18:49:39
npm http request GET https://us.registry.nodejitsu.com/mute-stream
npm http 304 https://us.registry.nodejitsu.com/revalidator
npm http 304 https://us.registry.nodejitsu.com/lru-cache
npm http 304 https://us.registry.nodejitsu.com/event-stream
npm http 304 https://us.registry.nodejitsu.com/mute-stream
npm http 304 https://us.registry.nodejitsu.com/ground
Package Current Latest at st Location
colors 0.6.0 0.6.2 0.6.3 colors
nssocket 0.5.1 0.5.2 0.5.2 nssocket
vows 0.7.0 0.7.0 0.8.0 vows
request 2.47.0 2.48.0 2.48.0 request
colors 0.6.2 0.6.2 1.0.3 cliff > colors
optimist 0.6.0 0.6.0 0.6.1 node > optimist
async 0.2.0 0.2.9 0.9.0 colors > async
minimist 0.0.10 0.0.10 1.1.0 optimist > minimalist
async 0.2.10 0.2.10 0.9.0 utile > async
minimist 0.0.8 0.0.8 1.1.0 utile > mkdirp > minimalist
ncp 0.4.2 0.4.2 1.0.1 utile > ncp
colors 0.6.2 0.6.2 1.0.3 winston > colors
async 0.2.10 0.2.10 0.9.0 winston > async
winston 0.8.0 0.8.0 0.8.3 broadway > winston
optimist 0.6.0 0.6.0 0.6.1 flatiron > optimist
revalidator 0.1.0 0.1.0 0.3.0 flatiron > prompt > revalidator
```

```
npm http 304 https://us.registry.nodejitsu.com/stack-trace
npm http 304 https://us.registry.nodejitsu.com/eventemitter2
npm http 304 https://us.registry.nodejitsu.com/pkginfo
npm http 304 https://us.registry.nodejitsu.com/read
npm info attempt registry request try #1 at 18:49:39
npm http request GET https://us.registry.nodejitsu.com/mute-stream
npm http 304 https://us.registry.nodejitsu.com/revalidator
npm http 304 https://us.registry.nodejitsu.com/lru-cache
npm http 304 https://us.registry.nodejitsu.com/minimist
npm http 304 https://us.registry.nodejitsu.com/event-stream
npm http 304 https://us.registry.nodejitsu.com/mute-stream
npm http 304 https://us.registry.nodejitsu.com/minimist
Package          Current  Wanted  Latest  Location
colors           0.6.2    0.6.2   1.0.3   colors
nssocket         0.5.1    0.5.2   0.5.2   nssocket
vows              0.7.0    0.7.0   0.8.0   vows
request          2.47.0   2.48.0  2.48.0   request
colors           0.6.2    0.6.2   1.0.3   cliff > colors
optimist          0.6.0    0.6.0   0.6.1   nconf > optimist
async             0.2.9    0.2.9   0.9.0   nconf > async
minimist          0.0.10   0.0.10  1.1.0   optimist > minimist
async             0.2.10   0.2.10  0.9.0   utile > async
minimist          0.0.8    0.0.8   1.1.0   utile > mkdirp > minimist
ncp               0.4.2    0.4.2   1.0.1   utile > ncp
colors           0.6.2    0.6.2   1.0.3   winston > colors
async             0.2.10   0.2.10  0.9.0   winston > async
winston            0.8.0    0.8.0   0.8.3   broadway > winston
optimist          0.6.0    0.6.0   0.6.1   flatiron > optimist
revalidator       0.1.2    0.1.2   0.3.0   flatiron > prompt > revalidator
```

NPM OUTDATED

IS A PURE GRAPH QUESTION



PEOPLE
WHO DEPEND ON X
— ALSO —
DEPEND ON WHAT OTHER MODULES?

dependencies: 1272 of 5330

devDependencies: 127 of 261

— IS THIS MODULE —
USED MORE IN PRODUCTION
— ALSO —
OR IN DEVELOPMENT?

— WHAT IS THE —
“**most stable**” VERSION
— OF —
THIS MODULE?

Analysis of **winston**

go to: [codependencies](#) [methods](#)

SVG edges clipped? Best viewed in Firefox - [see bug](#).

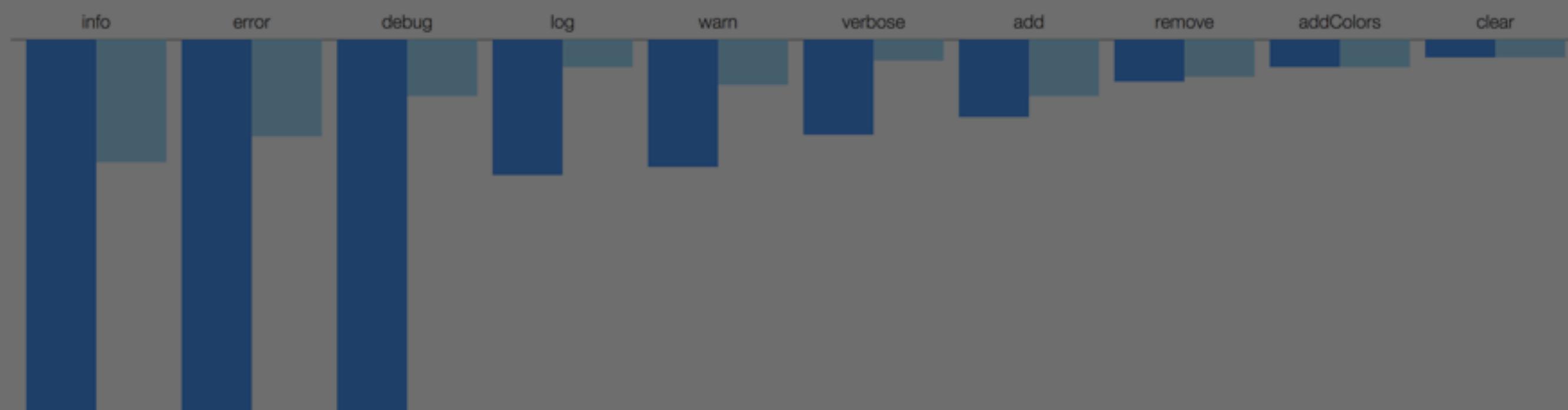
Codependency Graphs

OTHER QUESTIONS NEED
STATIC ANALYSIS
TO BE ANSWERED

dependencies: 1272 of 5350

DevDependencies: 127 of 261

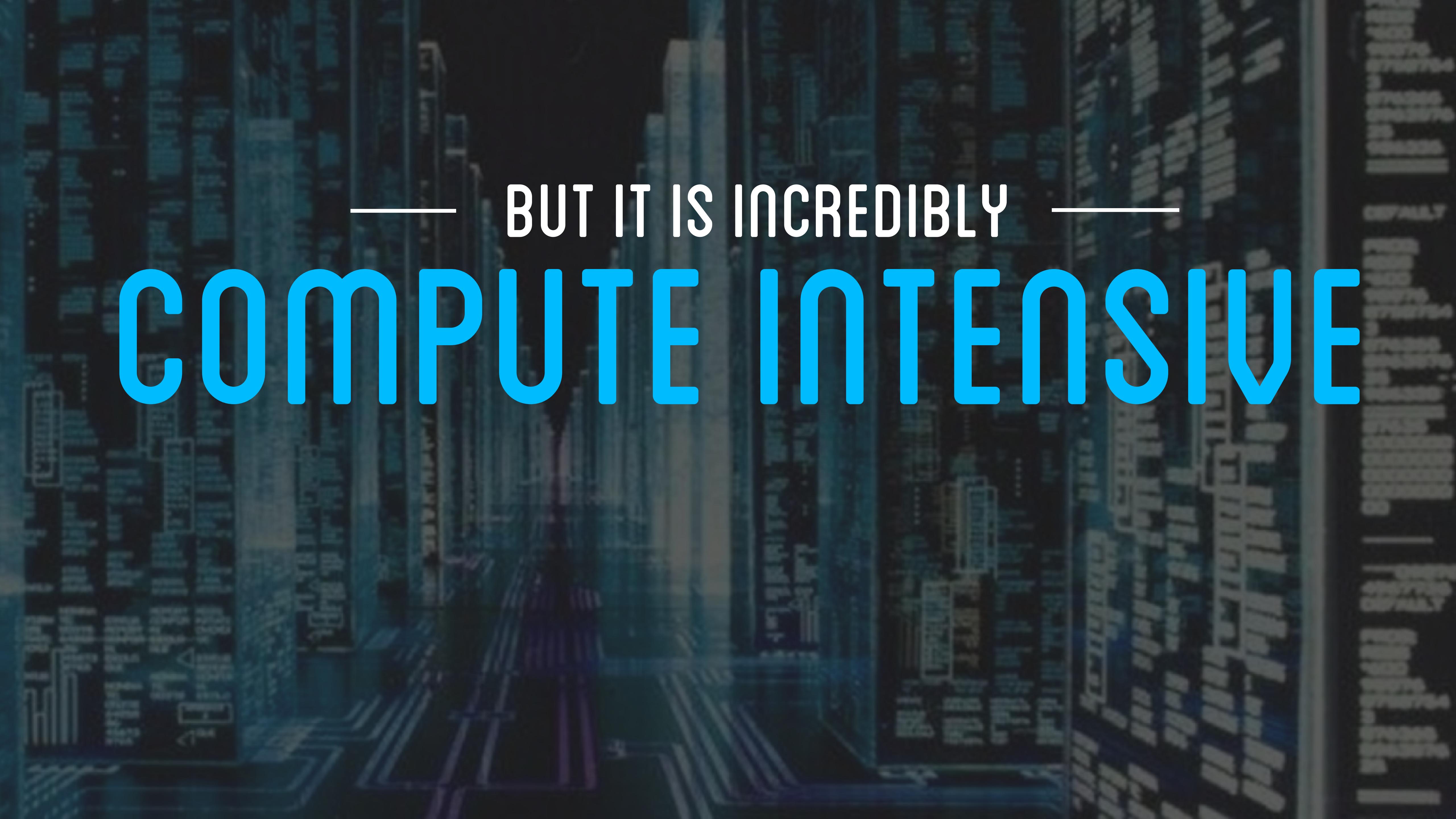
Most used methods



— HOW OFTEN —
DO OTHER MODULES
— USE —
A PARTICULAR METHOD?

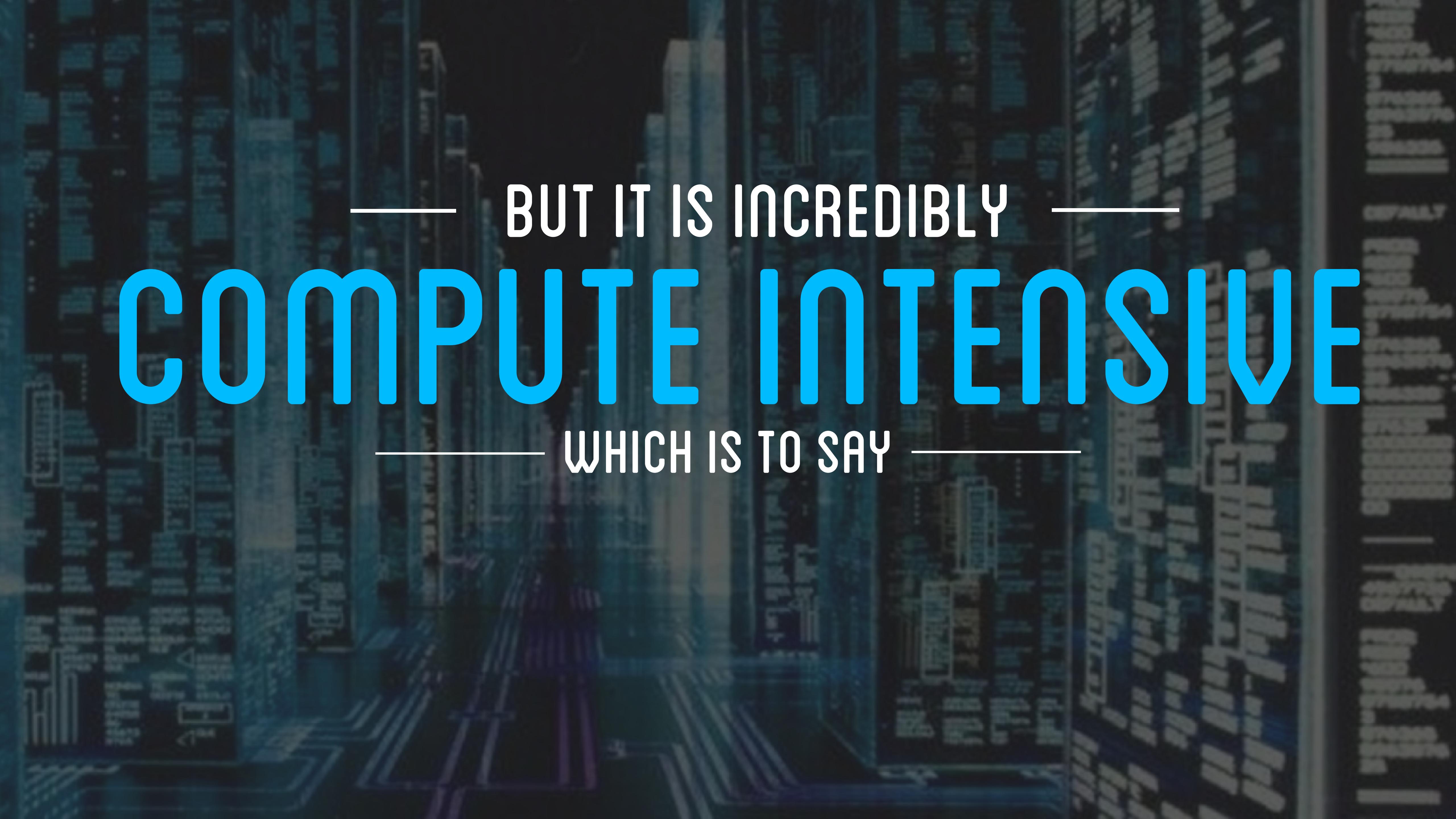
— HOW MANY MODULES —
ARE VULNERABLE
— TO —
SHELLSHOCK?

— CAN MY APP —
BE UPGRADED TO
— USE —
VERSION X.Y.Z SAFELY?



— BUT IT IS INCREDIBLY —

COMPUTE INTENSIVE



— BUT IT IS INCREDIBLY —

COMPUTE INTENSIVE

— WHICH IS TO SAY —



— BUT IT IS INCREDIBLY —
COMPUTE INTENSIVE
— WHICH IS TO SAY —
PRETTY SLOW

```
npm http 304 https://us.registry.nodejitsu.com/eventemitter2
npm http 304 https://us.registry.nodejitsu.com/pkginfo
npm http 304 https://us.registry.nodejitsu.com/read
npm info attempt registry request try #1 at 18:49:39
npm http request GET https://us.registry.nodejitsu.com/mute-stream
npm http 304 https://us.registry.nodejitsu.com/revalidator
npm http 304 https://us.registry.nodejitsu.com/lru-cache
npm http 304 https://us.registry.nodejitsu.com/minimist
npm http 304 https://us.registry.nodejitsu.com/event-stream
npm http 304 https://us.registry.nodejitsu.com/mute-stream
npm http 304 https://us.registry.nodejitsu.com/sigmund
```

<u>Package</u>	<u>Current</u>	<u>Wanted</u>	<u>Latest</u>	<u>Location</u>
colors	0.6.2	0.6.2	1.0.3	colors
nssocket	0.5.1	0.5.2	0.5.2	nssocket
vows	0.7.0	0.7.0	0.8.0	vows
request	2.47.0	2.48.0	2.48.0	request
colors	0.6.2	0.6.2	1.0.3	cliff > colors
optimist	0.6.0	0.6.0	0.6.1	nconf > optimist
async	0.2.9	0.2.9	0.9.0	nconf > async
minimist	0.0.10	0.0.10	1.1.0	optimist > minimist
async	0.2.10	0.2.10	0.9.0	utile > async
minimist	0.0.8	0.0.8	1.1.0	utile > mkdirp > minimist
ncp	0.4.2	0.4.2	1.0.1	utile > ncp
colors	0.6.2	0.6.2	1.0.3	winston > colors
async	0.2.10	0.2.10	0.9.0	winston > async
winston	0.8.0	0.8.0	0.8.3	broadway > winston
optimist	0.6.0	0.6.0	0.6.1	flatiron > optimist
revalidator	0.1.8	0.1.8	0.3.0	flatiron > prompt > revalidator

```
npm http 304 https://us.registry.nodejitsu.com/eventemitter2
npm http 304 https://us.registry.nodejitsu.com/pkginfo
npm http 304 https://us.registry.nodejitsu.com/read
npm info attempt registry request try #1 at 18:49:39
npm http request GET https://us.registry.nodejitsu.com/mute-stream
npm http 304 https://us.registry.nodejitsu.com/revalidator
npm http 304 https://us.registry.nodejitsu.com/lru-cache
npm http 304 https://us.registry.nodejitsu.com/minimist
npm http 304 https://us.registry.nodejitsu.com/overhead
npm http 304 https://us.registry.nodejitsu.com/mute-stream
npm http 304 https://us.registry.nodejitsu.com/big-unix
npm http 304 https://us.registry.nodejitsu.com/big-unix
Package Current Wanted Latest Location
colors      0.6.2  0.6.2  1.0.3  colors
nssocket    0.5.1  0.5.2  0.5.2  nssocket
vows        0.7.0  0.7.0  0.8.0  vows
request     2.47.0 2.40.0 2.40.0  request
colors      0.6.2  0.6.2  1.0.3  cliff > colors
optimist    0.6.0  0.6.0  0.6.1  optimist
async       0.2.9  0.2.9  0.3.0  async
minimist    0.0.10 0.0.10 0.0.11  minimist
async       0.2.10 0.2.10 0.2.10  util > async
minimist    0.0.8  0.0.8  1.1.0  util > mkdirp > minimist
ncp         0.4.2  0.4.2  1.0.1  util > ncp
colors      0.6.2  0.6.2  1.0.3  winston > colors
async       0.2.10 0.2.10 0.9.0  winston > async
winston     0.8.0  0.8.0  0.8.3  broadway > winston
optimist    0.6.0  0.6.0  0.6.1  flatiron > optimist
revalidator 0.1.8  0.1.8  0.3.0  flatiron > prompt > revalidator
```

NPM OUTDATED

IS REALLY FAST. —
THAT'S HOW YOU CAN USE IT —

EVERY DAY

— BY USING —



— OF COURSE —

YO DAWG I HEARD YOU LIKE npm

— BY USING —



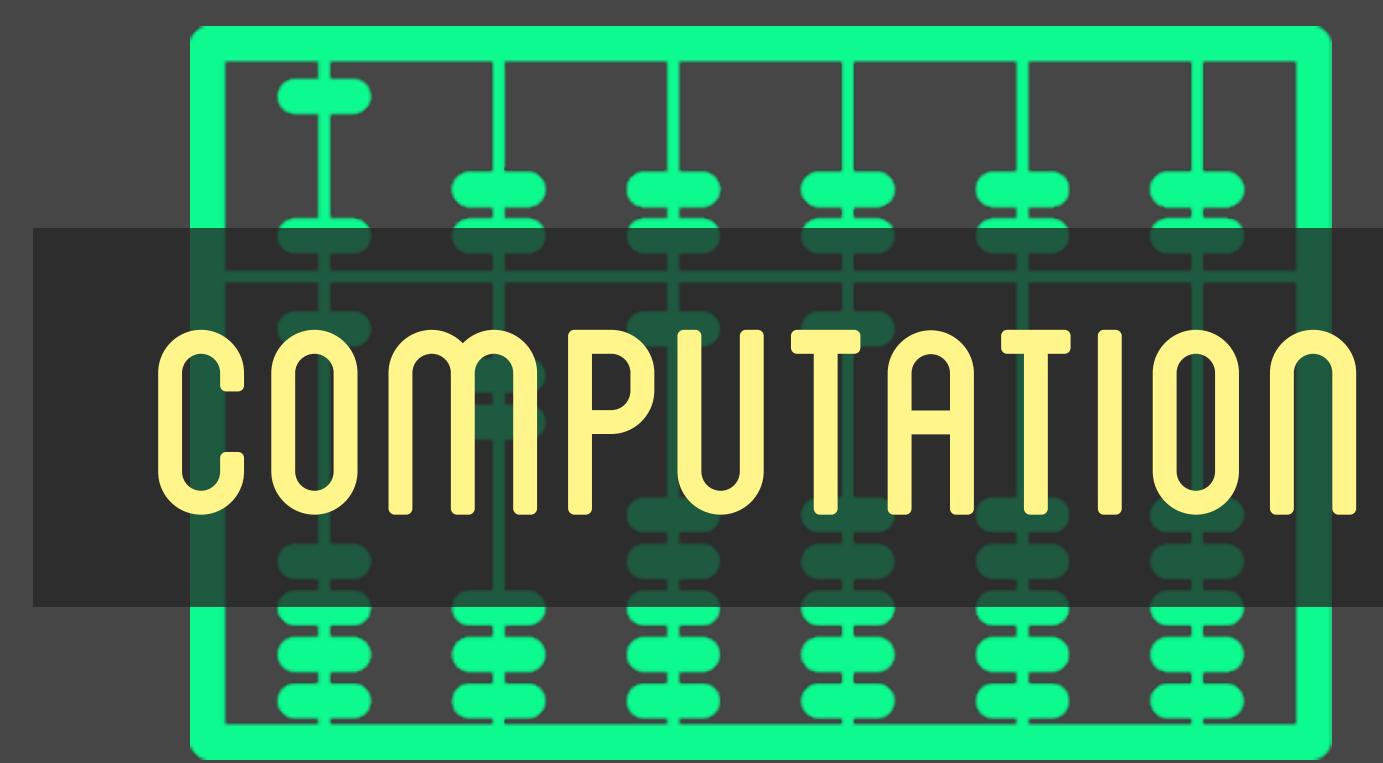
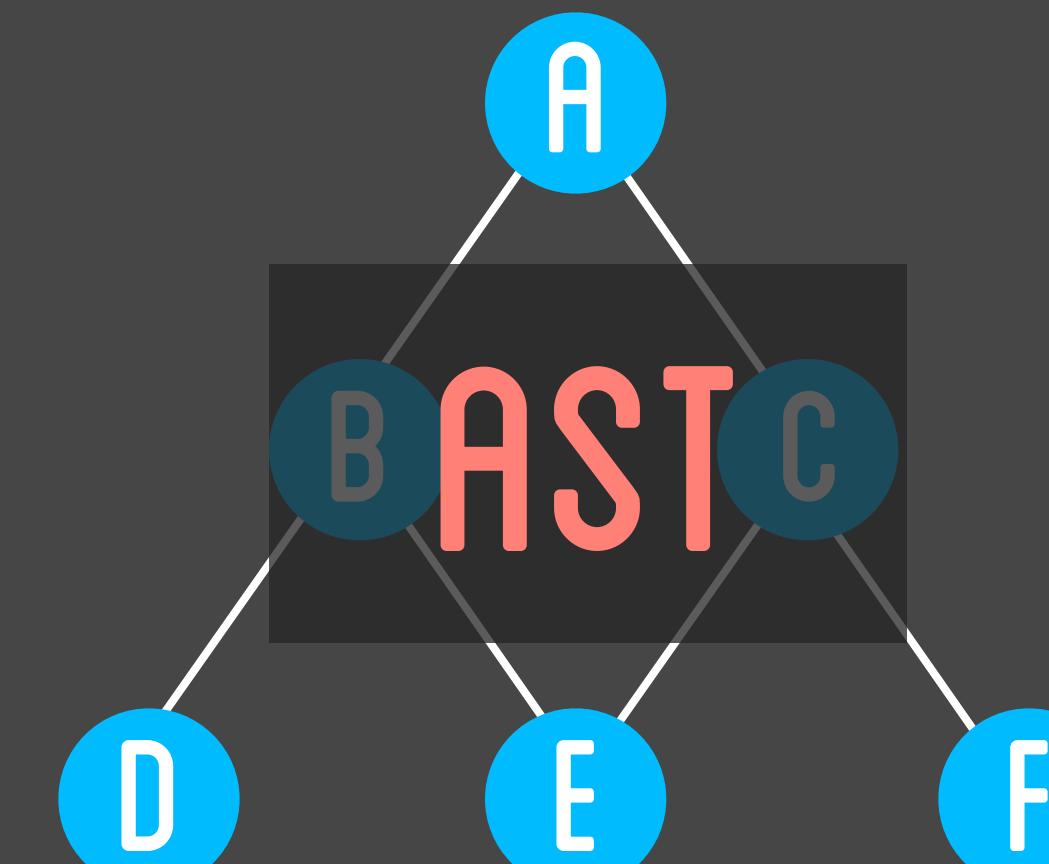
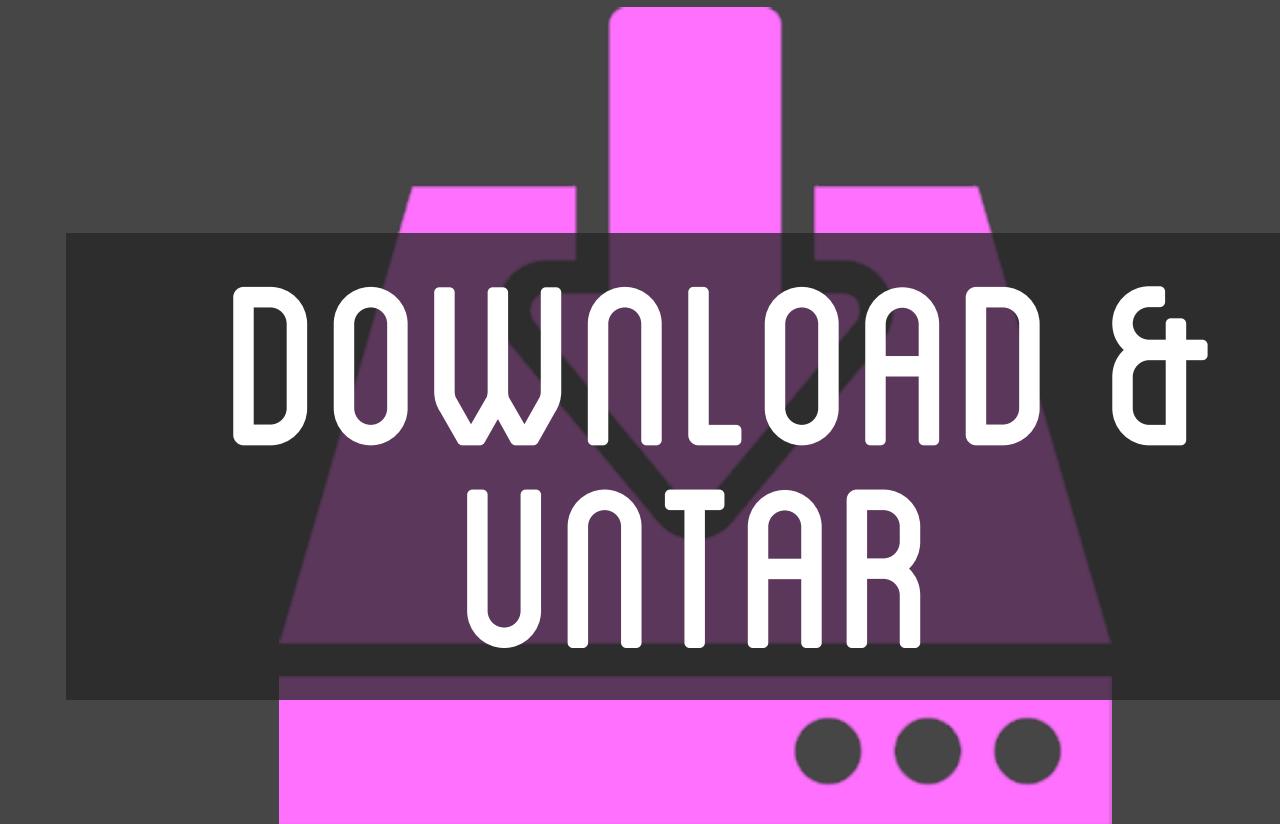
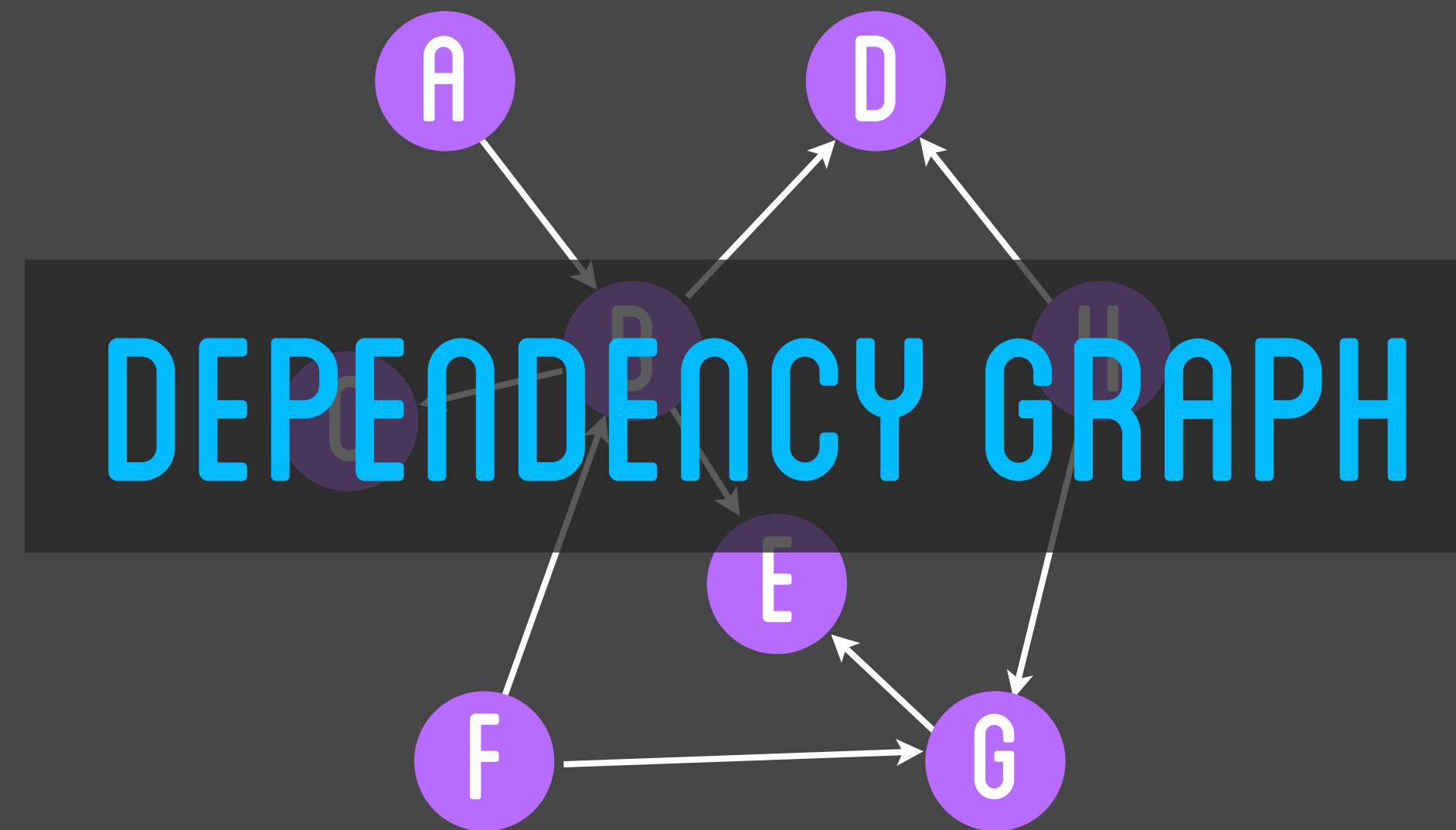
— OF COURSE —

ALONG WITH

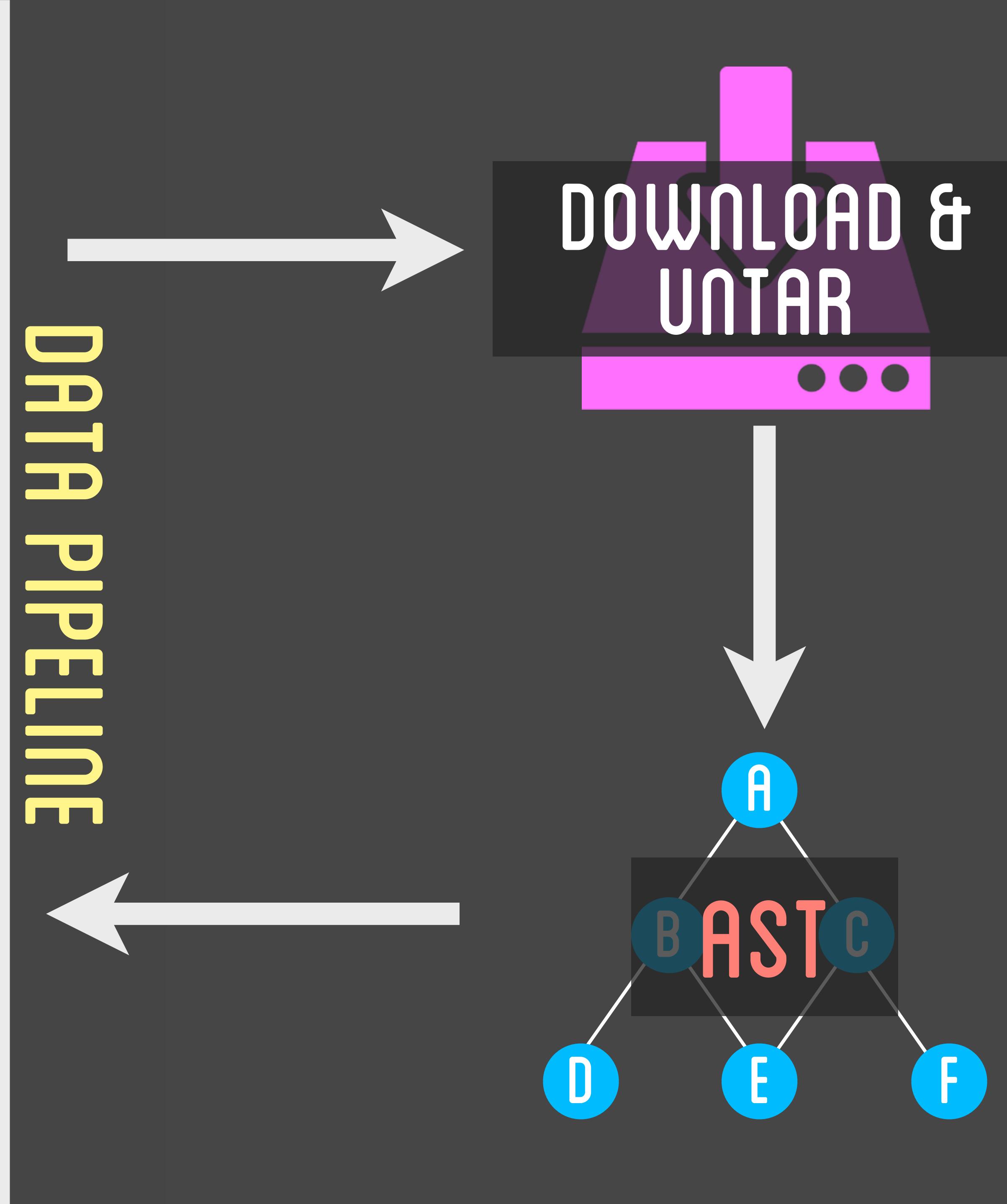
— A —

DATA PIPELINE

YO DAWG I HEARD YOU LIKE npm



THE SET OF
MODULES
AND THE SPECIFIC
ANALYSIS WORK
CHANGES BUT, THE
DATA PIPELINE
STAYS THE SAME
MAKING IT
HIGHLY PARALLELIZABLE



ESPRIMA + RECAST

+ NPM + DATA PIPELINE

= NPM-PIPELINE

```
#!/usr/bin/env node

var path = require('path'),
  pipeline = require('npm-pipeline'),
  argv    = require('yargs').argv;

//  

// ### function analyzeModuleMap (name, next)  

// Pipelines the module with `name` and then  

// respond with the results to be reduced.  

//  

function analyzeModuleMap(name, next) {
  pipeline(name, function (err, files) {
    if (err) { return next(err); }

    var results;
    try { results = require(path.resolve(argv.work))(files, argv); }
    catch (ex) { return next(ex); }

    next(null, results);
  });
}

analyzeModuleMap(argv.module, function (err, res) {
  if (err) { return console.error(err); }
  process.stdout.write(JSON.stringify(res));
});
```

```
#!/usr/bin/env node

var path = require('path'),
  pipeline = require('npm-pipeline'),
  argv    = require('yargs').argv;

//  
// ### function analyzeModuleMap (name, next)  
// Pipelines the module map name and then  
// respond with the results to be used  
//  
function analyzeModuleMap(name, next) {
  pipeline(name, function (err, files) {
    if (err) { return next(err); }

    var results;
    try { results = require('path').resolve(argv.root)(files, argv); }
    catch (ex) { return next(ex); }

    next(null, results);
  });
}

analyzeModuleMap(argv.module, function (err, res) {
  if (err) { return console.error(err); }
  process.stdout.write(JSON.stringify(res));
});
```

— A —
GENERIC MODULE

— AND —
GENERIC WORK TO PERFORM

— ANALYSIS TIME DOWN —

FROM TENS OF MINUTES

— TO —

TENS OF SECONDS

Codependency Graphs

WHAT OTHER QUESTIONS DO YOU HAVE FOR THE GRAPH?



THANKS

{GITHUB, TWITTER}.COM/INDEXZERO
INDEXZERO@NODEJITSU.COM