

Visualizing BEATOR2 Models

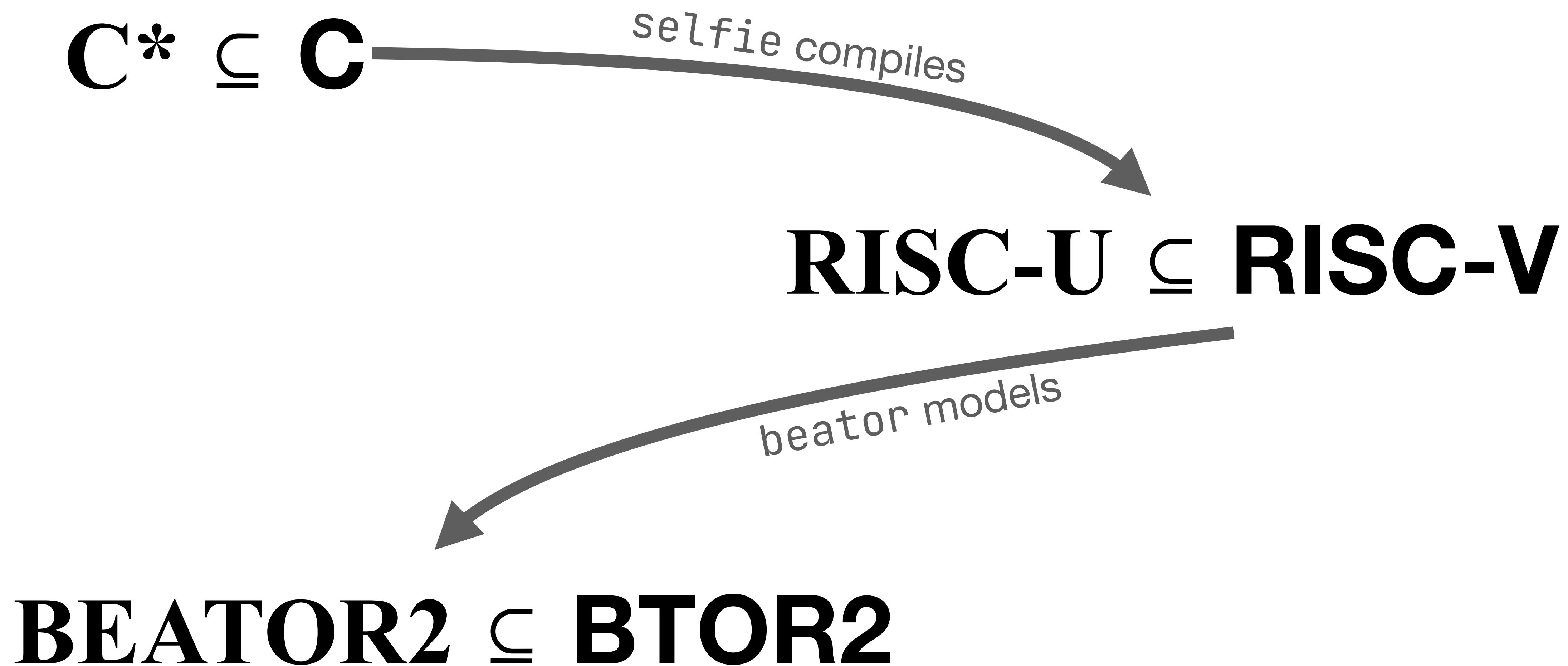
by Markus Diller

June 28, 2022

Advisor: Christoph Kirsch
Department of Computer Science

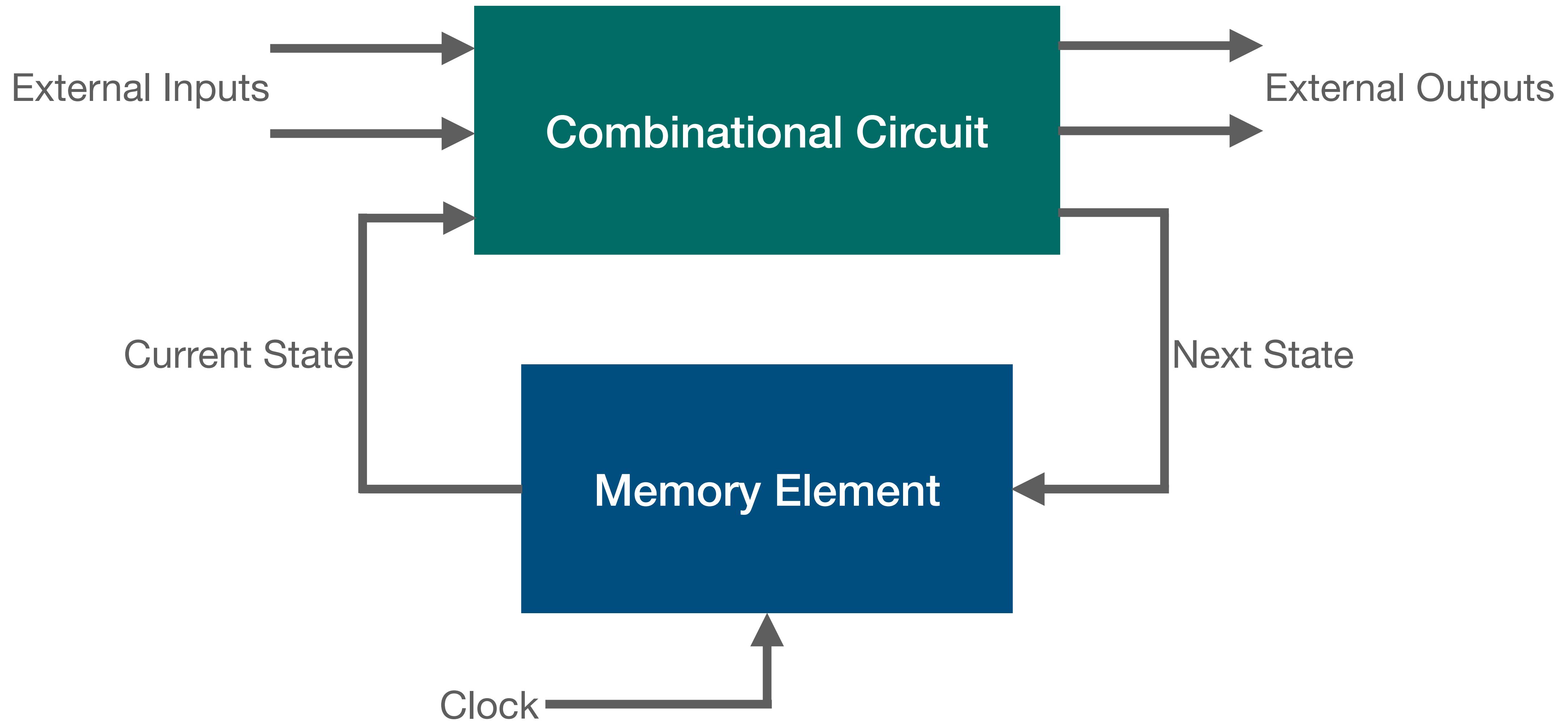
Introduction

C*, RISC-U, BEATOR2



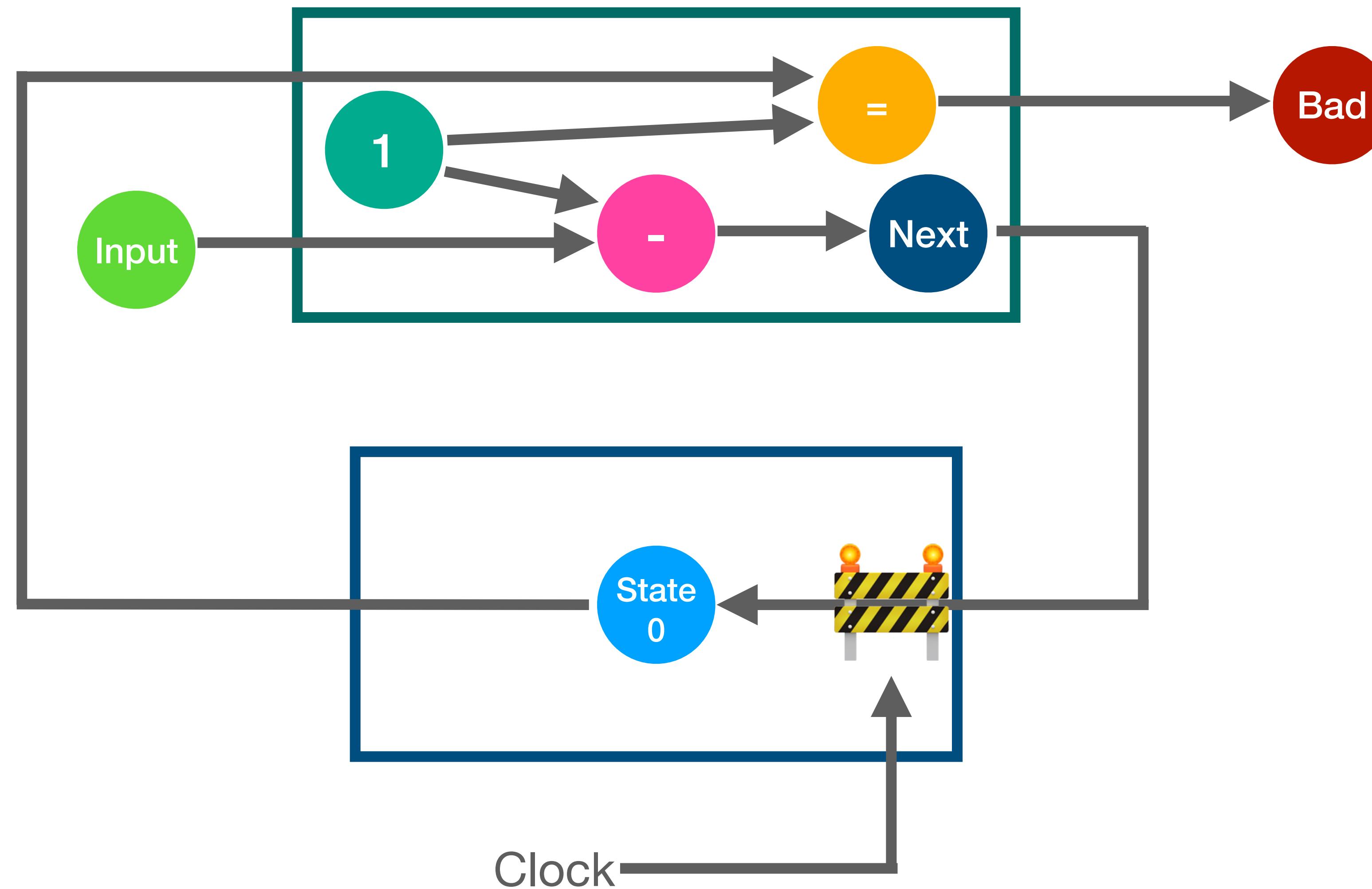
BEATOR2

Sequential Circuit



BEATOR2

BEATOR2 Circuit

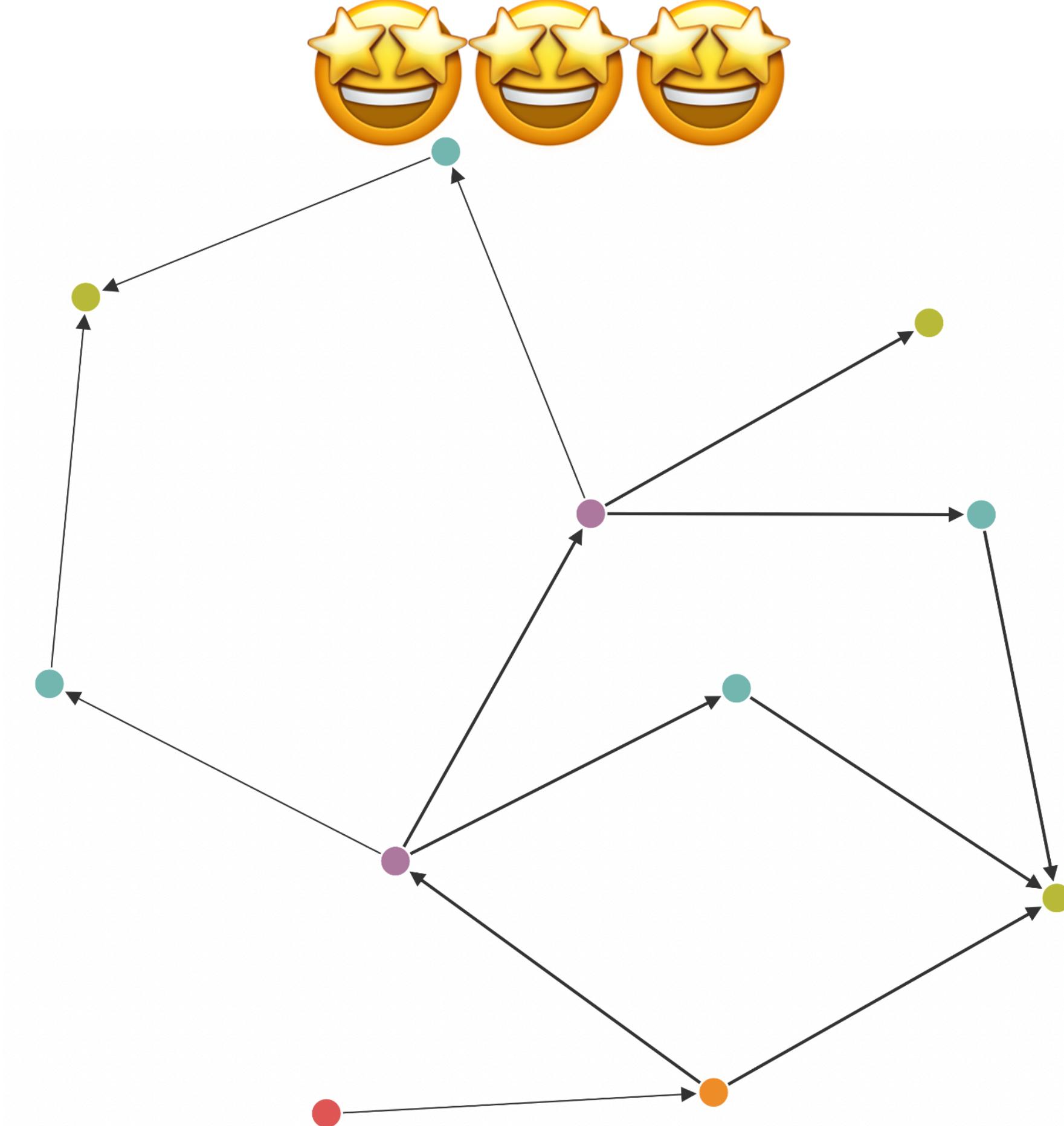


BEATOR2

Comparison



```
...
10000990 next 3 10000569 10000989
10000991 constd 2 56
10000992 eq 1 10000030 10000991
10000993 not 1 10000992
10000994 not 1 10000052
10000995 and 1 10000993 10000994
10000996 constd 2 64
10000997 eq 1 10000030 10000996
10000998 not 1 10000997
10000999 and 1 10000995 10000998
10001000 not 1 10000060
10001001 and 1 10000999 10001000
10001002 not 1 10000033
10001003 and 1 10001001 10001002
10001004 and 1 10000028 10001003
10001005 bad 10001004 invalid-syscall-id
10001006 eq 1 10000035 10000029
...
...
```



BEATOR2 Model File

Advantages and Disadvantages



No forward references

Node identifiers

Actual identifiers



Multiple classes for input

Separate initialization

Combinatorial and sequential part not
separable

Problem Definition

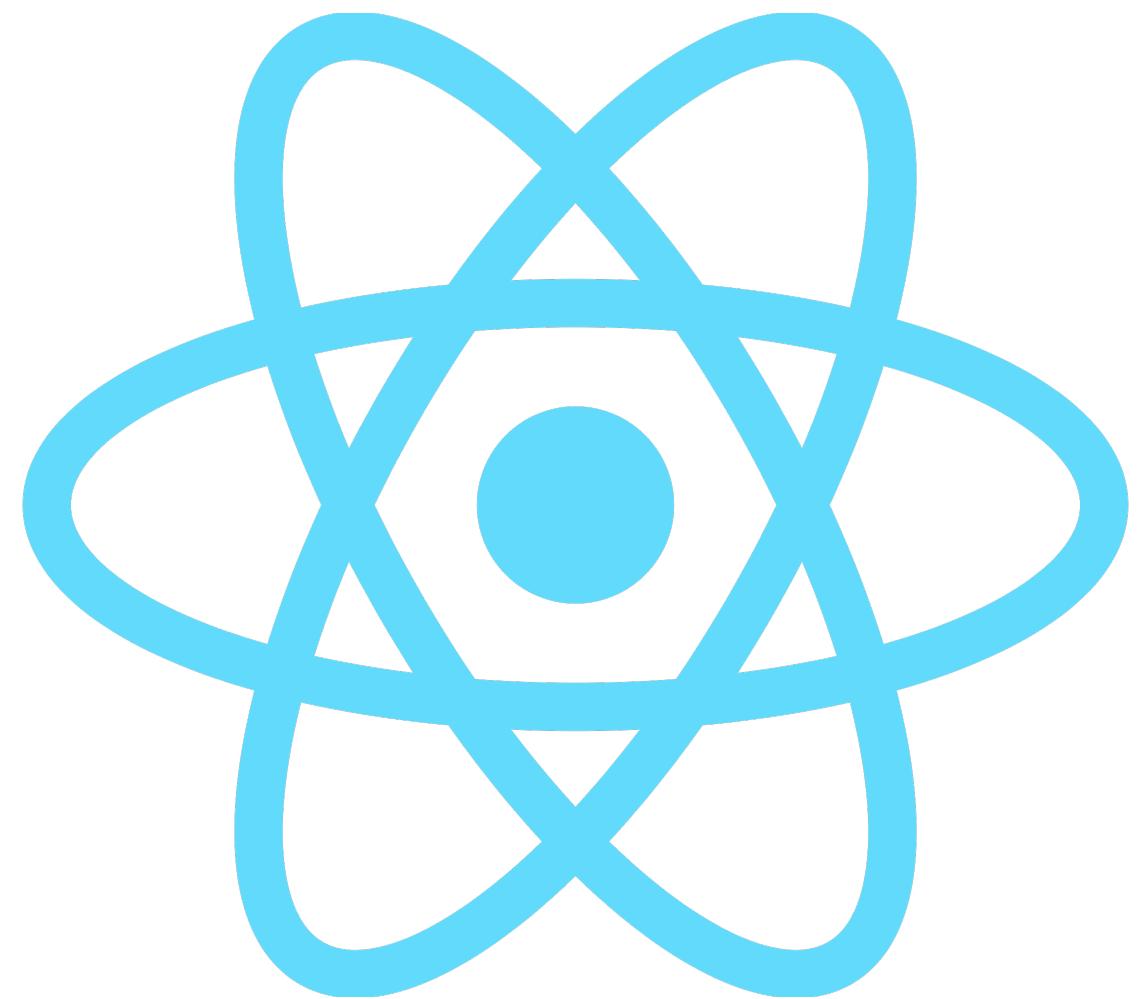
What I wanted to do

- Visually display the model in a graph
- Navigate paths through a model
- Provide information about a model
- Provide information about every node in a model

Used Technologies

Programming Environment

- React:
 - ▶ Easily create interactive web-applications
 - ▶ I was already familiar with it
 - ▶ Very popular
- TypeScript:
 - ▶ Don't have to keep track of types yourself
 - ▶ IntelliSense



Used Technologies

Visualization Library

- Data-Driven Documents (D3)
- Data visualization in web browsers
- Force Directed Graph
- Manipulate SVG elements



github.com/d3

Used Technologies

UI Library

- MUI/Material UI
- Implements Google's Material Design
- Less CSS Headaches
- Looks nicer than what I can do



mui.com

What have I done?

Calculations

Local Information

Local Information



NID: 10000065

Sort: Boolean

Name: kernel-mode

Class: Next

Depth: 0

Dependancy: 35



Constant: 5

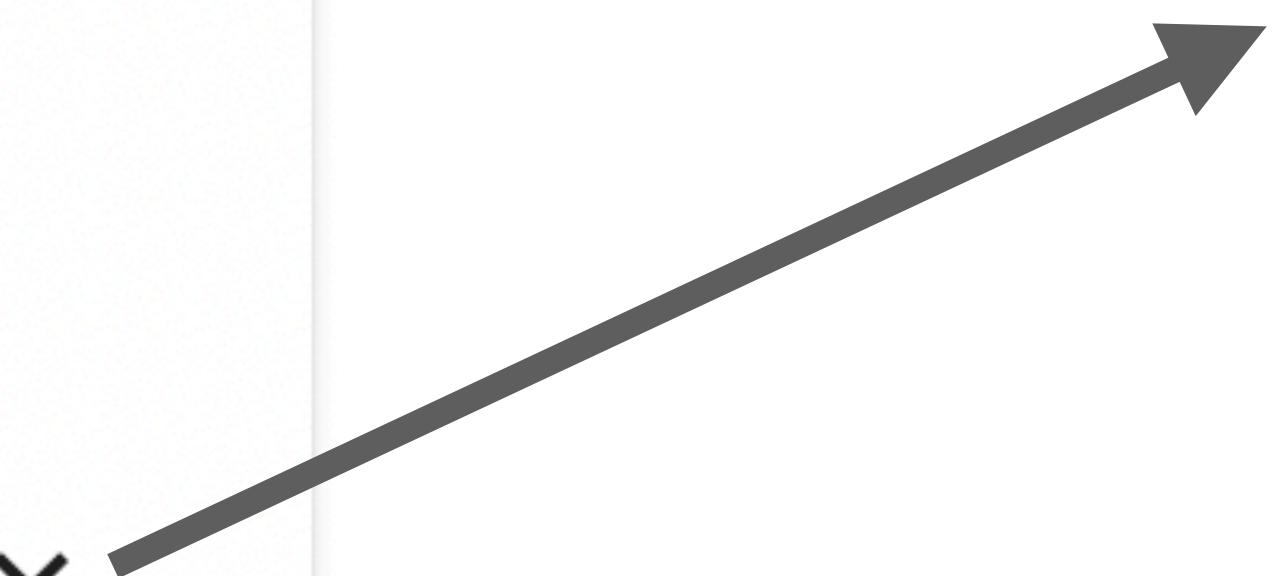
LessThan: 1

IfThenElse: 11

And: 4

Equals: 2

State: 12



Calculations

Global Information

Global Information

Size: 909 nodes

Bads: 10 bad instructions

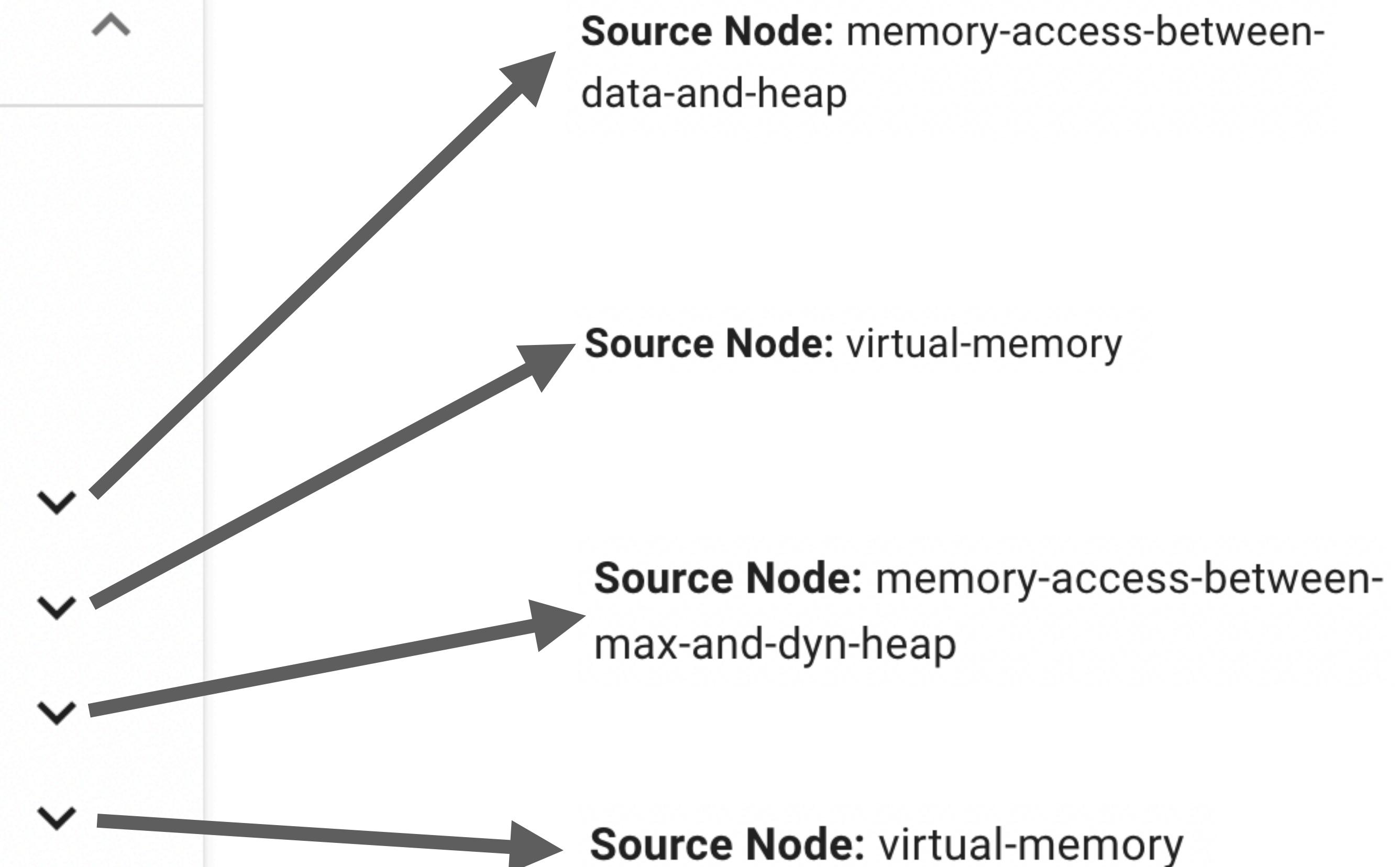
States: 184 PCs + registers

Longest Path: 51 nodes long

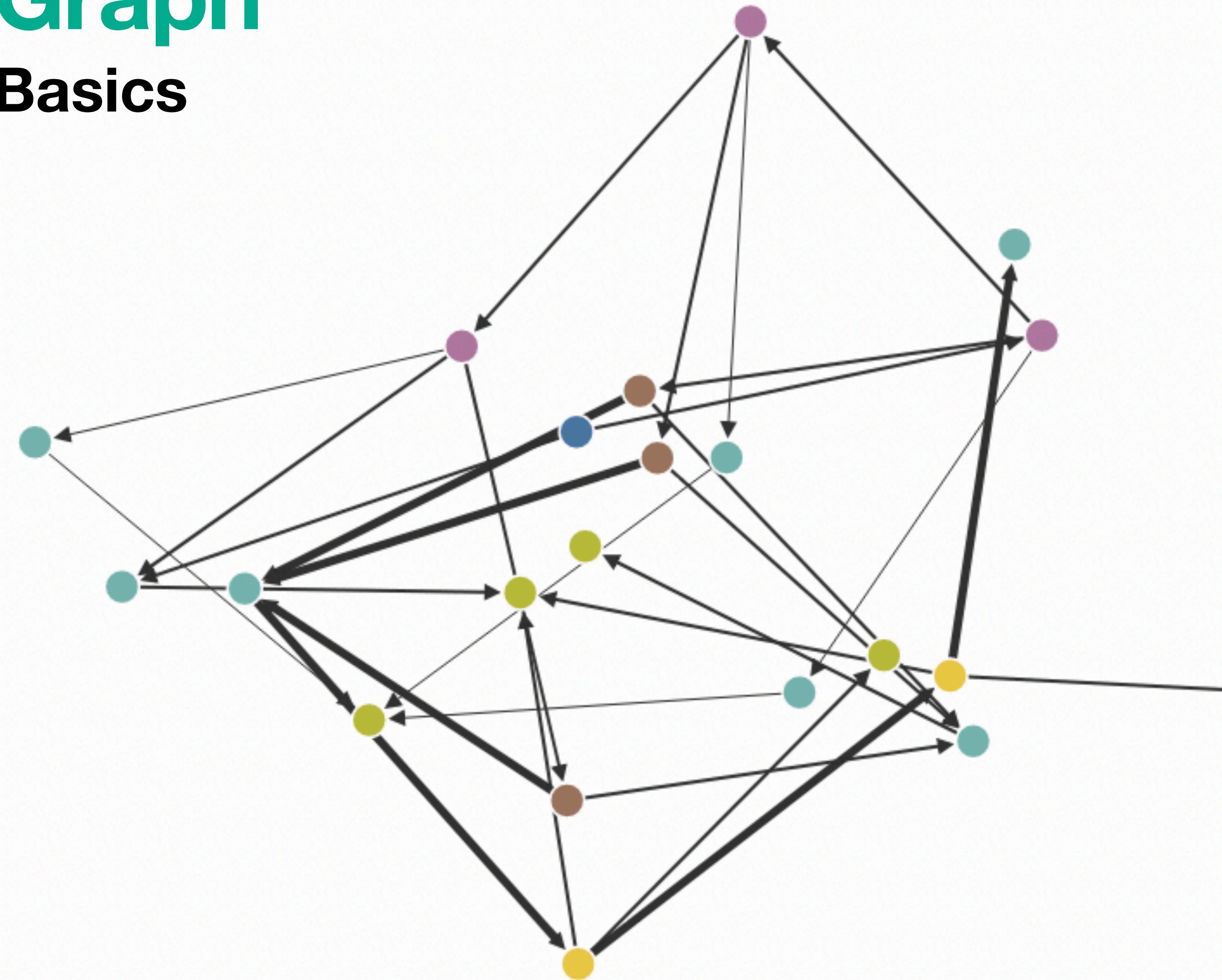
Longest Path from State: 3 nodes long

Max Dependancy: 144 nodes

Max Dependancy from State: 6 nodes



Graph Basics



Selection

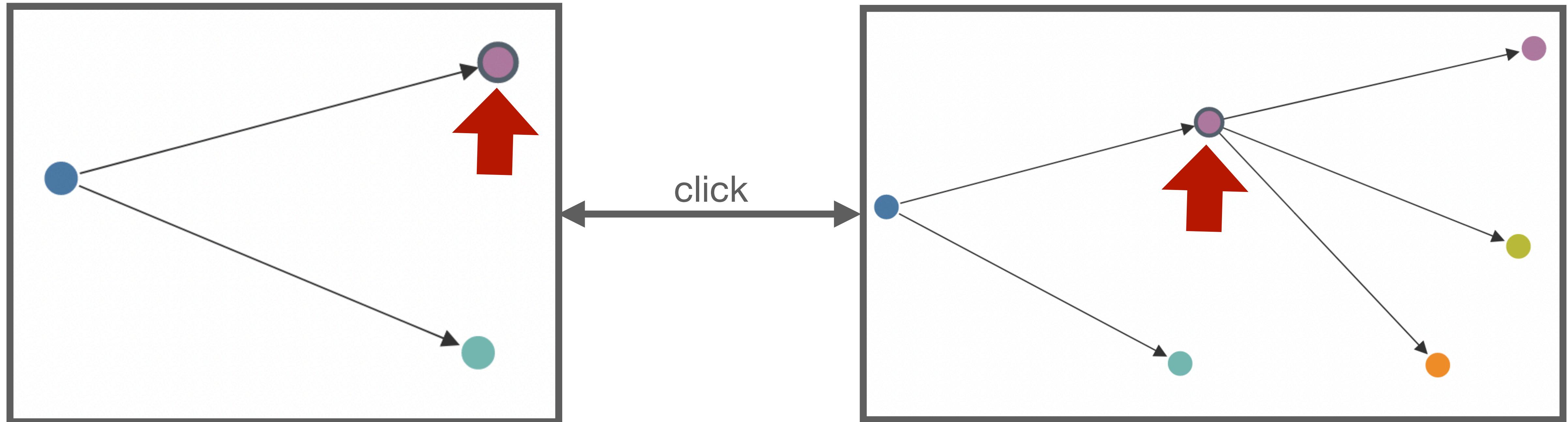
BADS	PCS	REGS	OTHER
division-by-zero			
invalid-syscall-id			
memory-access-above-stack			
memory-access-below-data			
memory-access-between-data-and-heap			
memory-access-between-dyn-and-max-stack			
memory-access-between-heap-and-stack			
memory-access-between-max-and-dvn-hean			

Menu

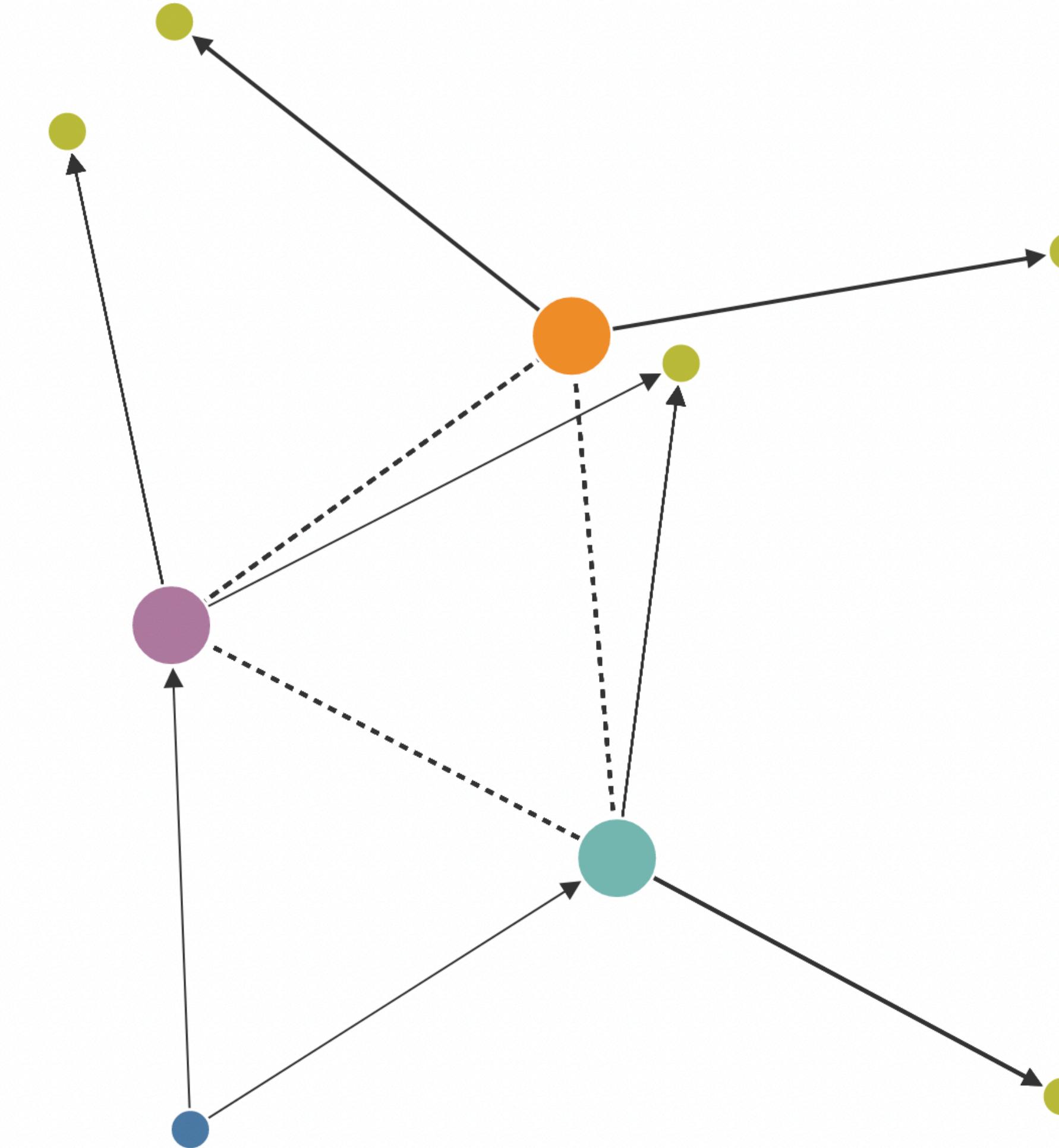
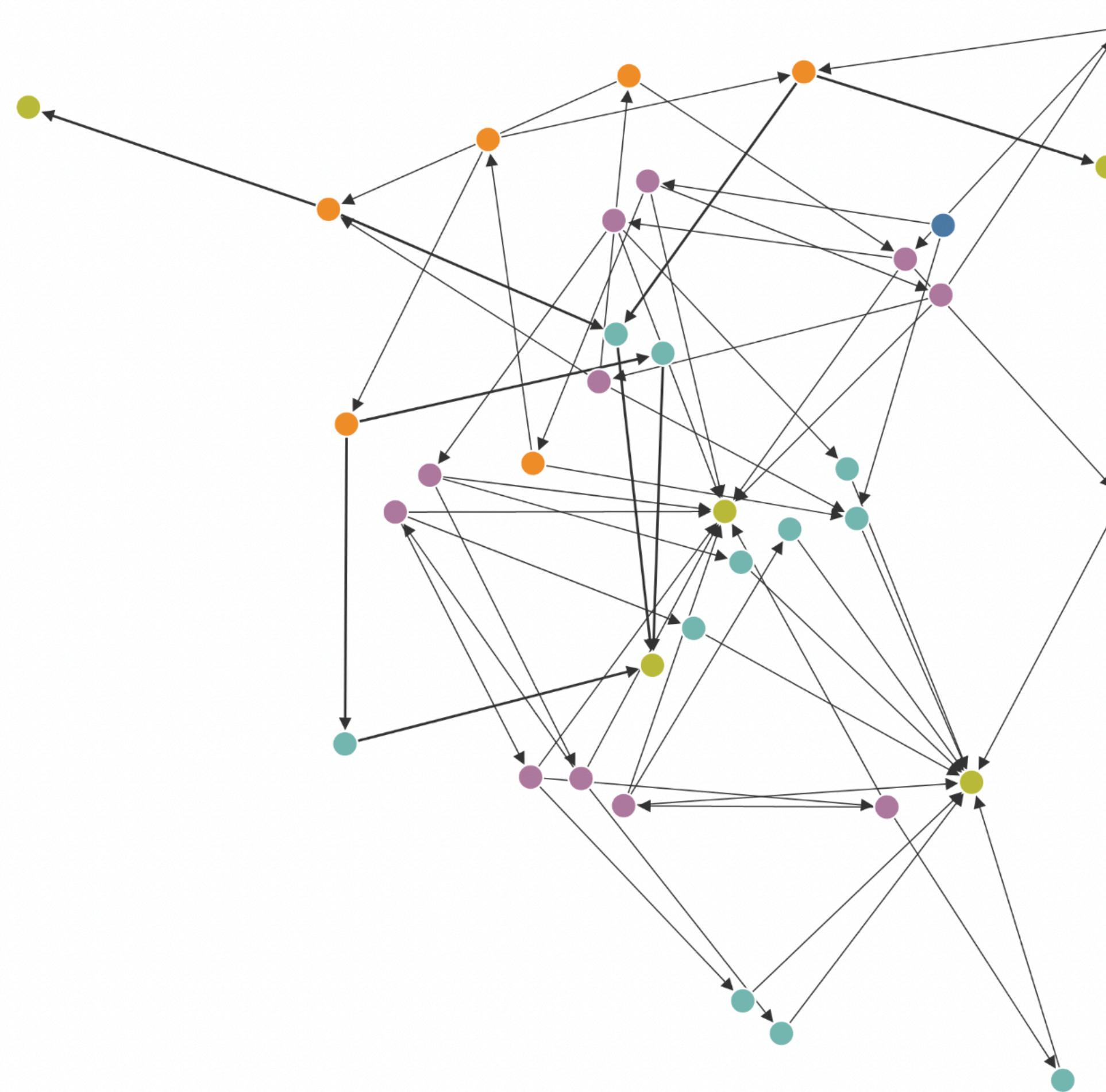
LEGEND	SETTINGS	MISC.
Bad		
Next		
Constant		
Input		
State		
And, Not, Equals, Less-than		
IfThenElse		
Addition, Subtraction, Division, Multiplication.		

Graph

Expand and Collapse

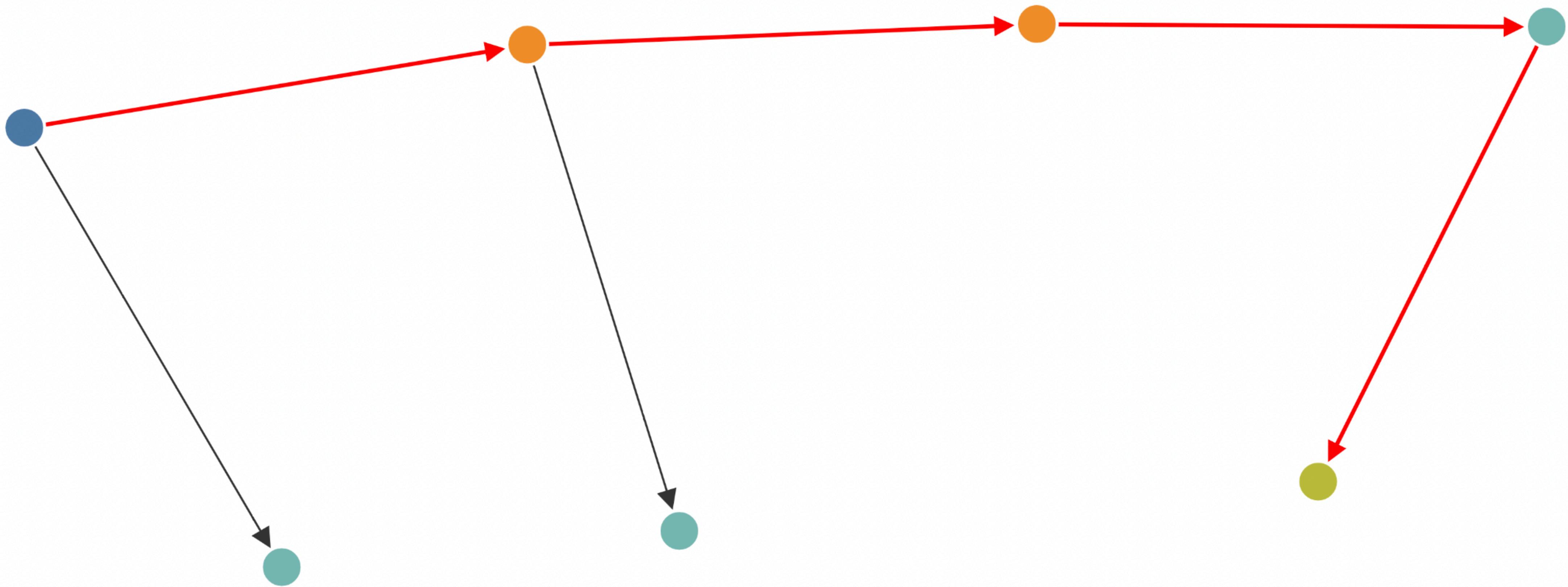


Graph Clumping



Graph

Longest Path

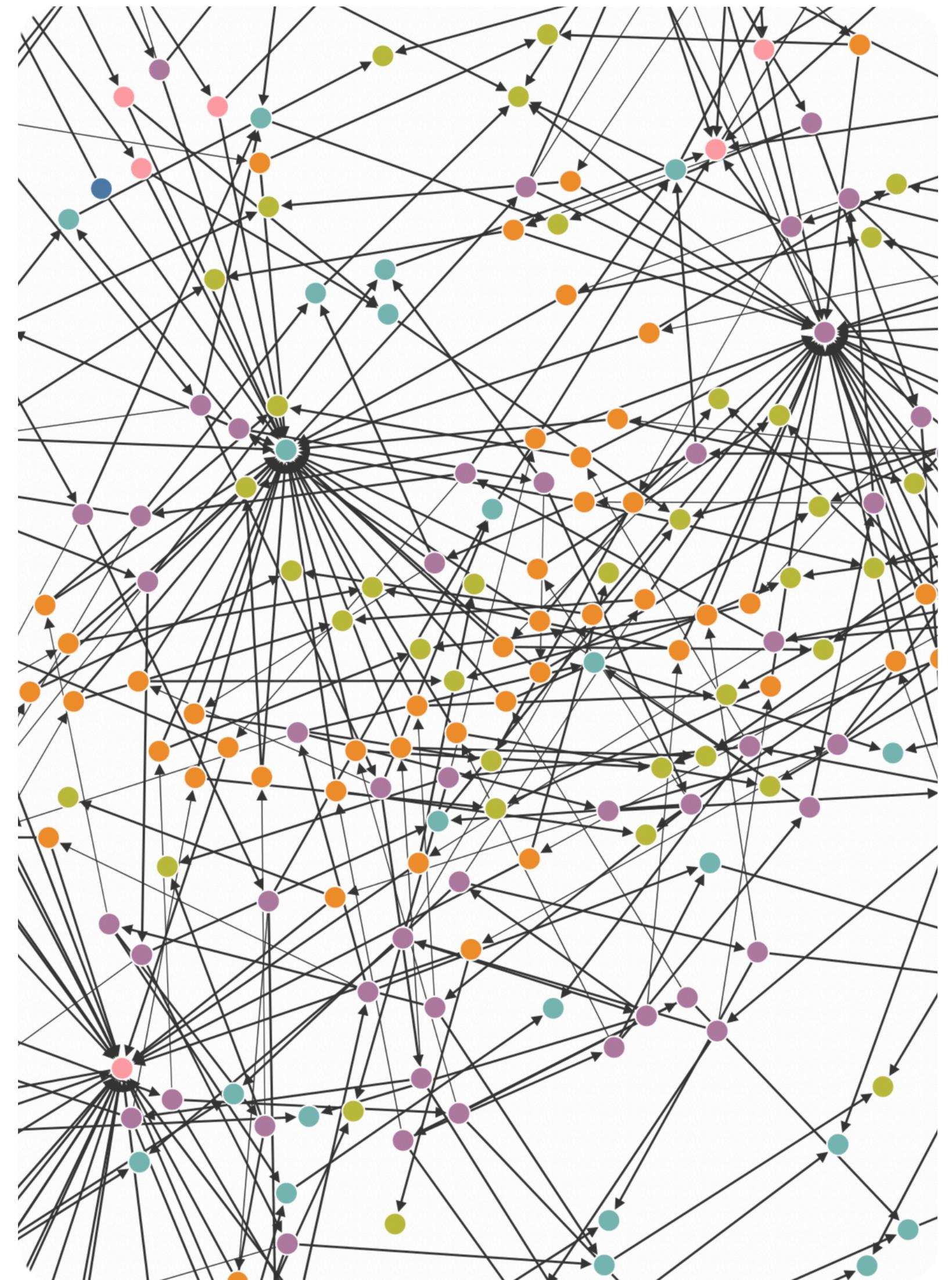


Conclusion Time!

Problems

Visualization/Interaction

- Bad scaling
- Suboptimal layouts
- Clutter
- Only left click

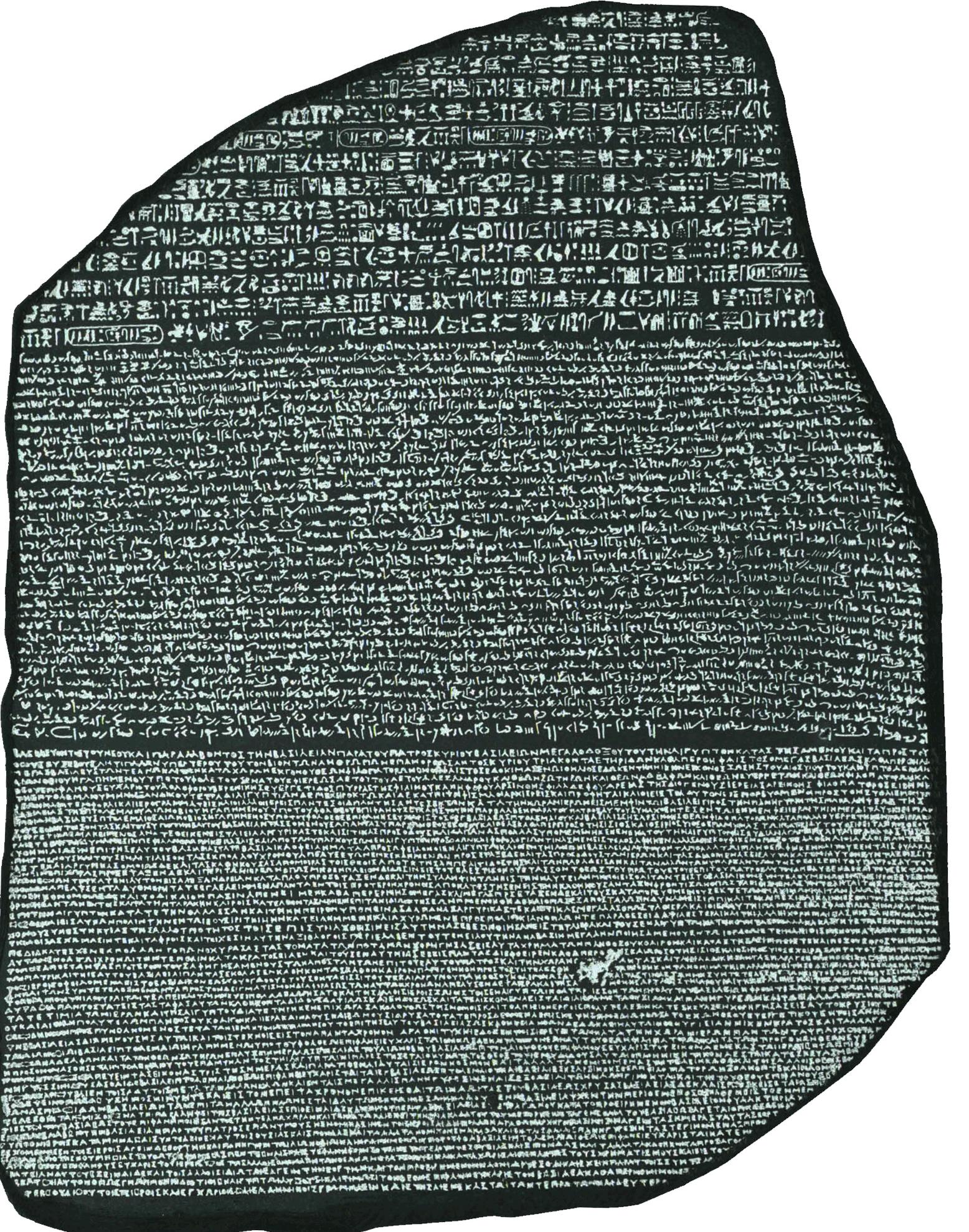


Problems

Programming Challenges

- Bad and outdated documentation
- Unwanted refreshes
- Prop Drilling and passing data
- D3 with React

Documentation be like:



Outlook

Possible Future Work

- Validate model file
- Emulate inputs
- Move to the third dimension
- Grow with the expansion of beator



medium.com

Thanks for listening!