

THE MISUNDERSTOOD ABSTRACTION

BY MARTIN CHAOV



AGENDA

What is abstraction?

What happens when abstractions fail?

What is the wrong level of abstraction?

What is the right level of abstraction?

Practical Advice

Summary

A fractal pattern of overlapping circles on a black background. The pattern is symmetrical and consists of a central circle with four smaller circles attached to its top, bottom, left, and right sides. Each of these four circles is further divided into four smaller circles, creating a complex, self-similar structure. The word "ABSTRACTION" is written in white, bold, sans-serif capital letters across the center of the image, partially overlapping the central circle.

ABSTRACTION



abstraction

/əb'strakʃ(ə)n/

noun

noun: **abstraction**

1. the quality of dealing with ideas rather than events.

"topics will vary in degrees of abstraction"

- something which exists only as an idea.

plural noun: **abstractions**

"the question can no longer be treated as an academic abstraction"

Similar:

concept

idea

notion

thought

generality

generalization



2. freedom from representational qualities in art.

"geometric abstraction has been a mainstay in her work"

- an abstract work of art.

"critics sought the meaning of O'Keeffe's abstractions"

3. a state of preoccupation.

"she sensed his momentary abstraction"

Similar:

absent-mindedness

distraction

preoccupation

daydreaming



4. the process of considering something independently of its associations or attributes.

"the question cannot be considered **in abstraction from** the historical context in which it was raised"

WHAT IS ABSTRACTION?

Natural language

Literature/Cinema/Music



Harry Potter

THE
MATRIX



STAR
WARS

FAVOURITE STORIES?

ABSTRACTION IS SPECIFIC

THE ESSENCE OF ABSTRACTIONS
IS PRESERVING INFORMATION
THAT IS RELEVANT IN A GIVEN
CONTEXT, AND FORGETTING
INFORMATION THAT IS
IRRELEVANT IN THAT CONTEXT.

SOFTWARE IS ABSTRACTION



WHAT HAPPENS WHEN
ABSTRACTIONS FAIL?



THE PRIMARY COST OF
ABSTRACTION IS **INDIRECTION.**

ONE MORE LEVEL OF

Tomcat / JBoss

Acegi

Spring MVC

Spring WebFlow

MVC Action

AOP TX Proxy

Business Logic

DAO

INDIRECTION

namegenerator.net

MARIO
058700

0x20

WORLD
1-4

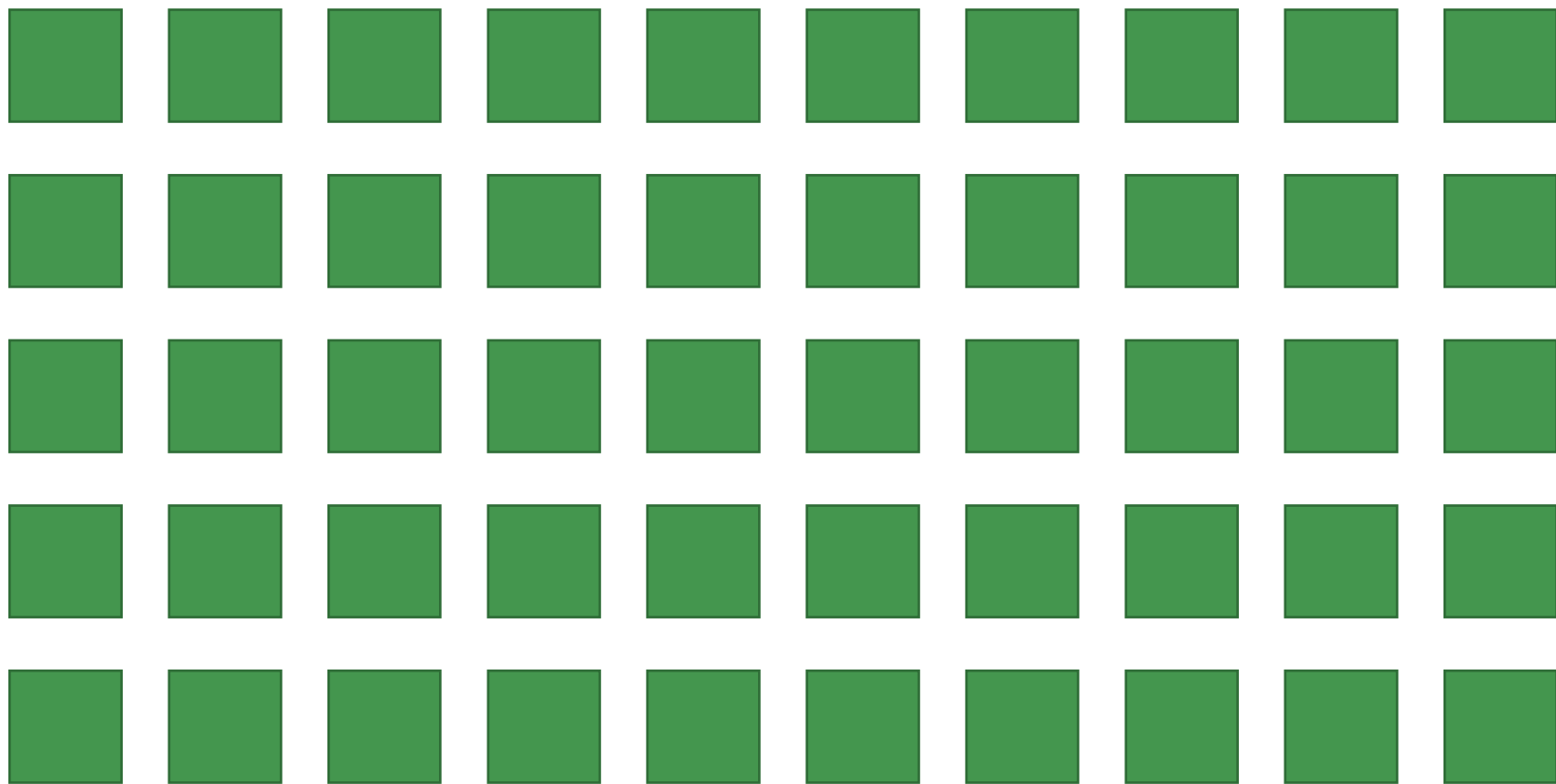
TIME
200

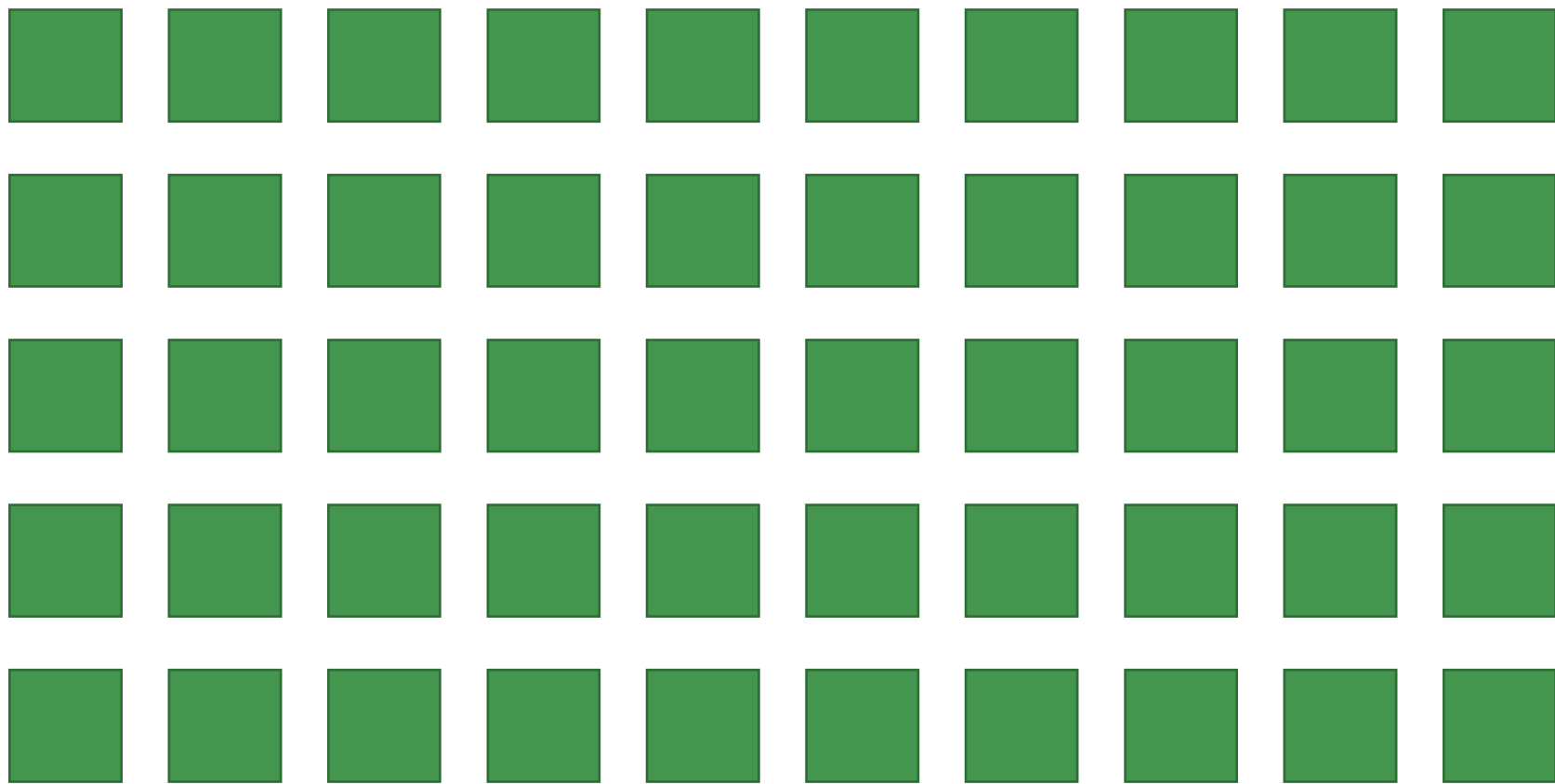
THANK YOU MARIO !

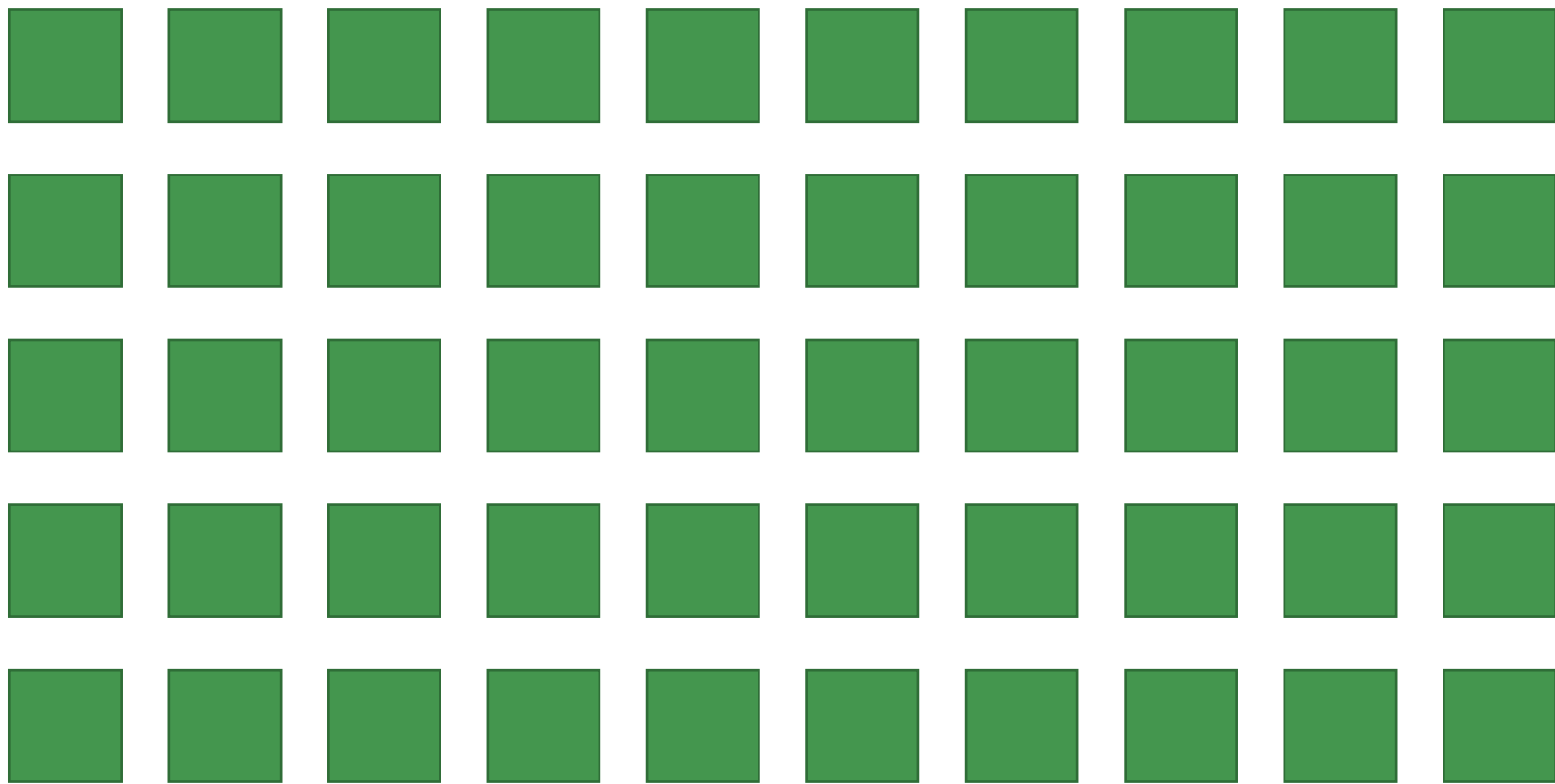
BUT THIS INTERFACE IMPLEMENTATION
IS IN ANOTHER PACKAGE...



LEAKY ABSTRACTIONS





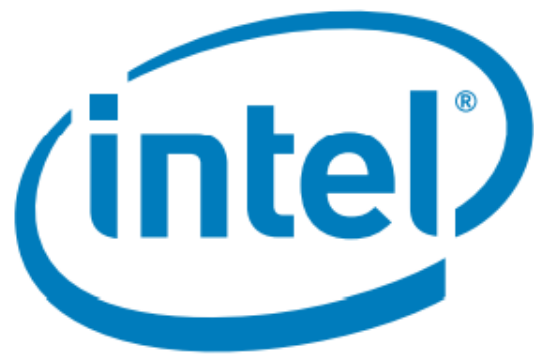


WHICH ONE IS THE FASTEST?

Loop by row

Loop by column

Loop by block



Intel® 64 and IA-32 Architectures Optimization Reference Manual

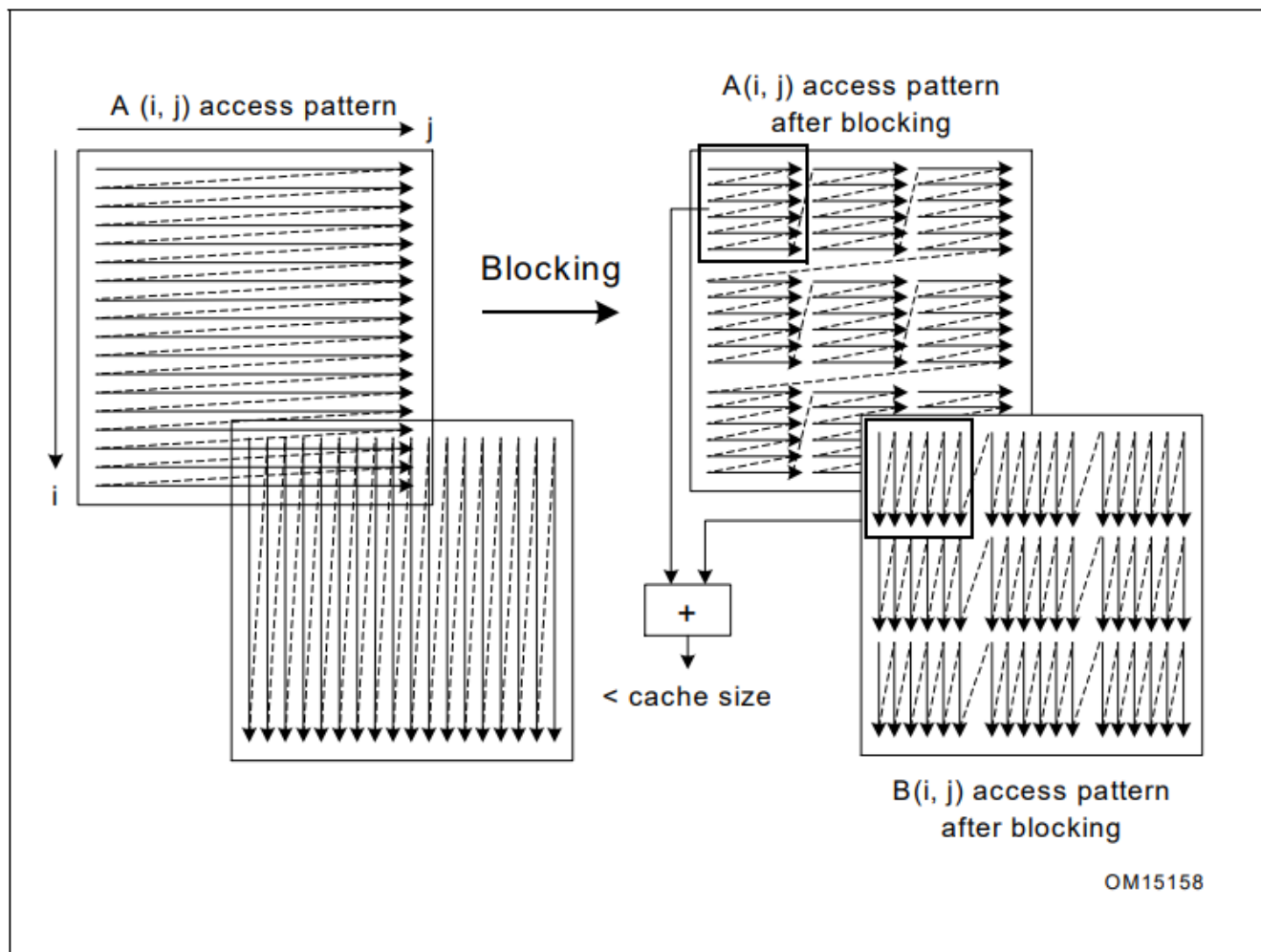


Figure 4-5. Loop Blocking Access Pattern

Your code is OK, you just don't know your CPUs



Loop Performance

The Definitive Guide

O RLY?

@raziel_pld

LAYERS OF ABSTRACTION

Electrons

Transistors

Logic gates

Circuits

Memory

CPU

OS

Browser

LAYERS OF ABSTRACTION

Electrons

Transistors

Logic gates

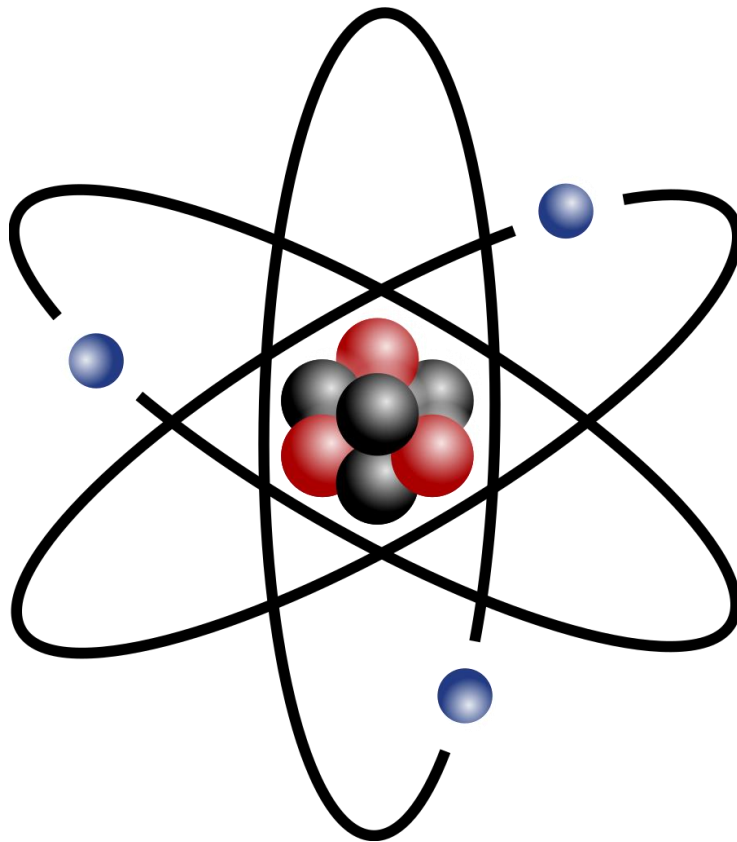
Circuits

Memory

CPU

OS

Browser



LAYERS OF ABSTRACTION

Electrons

Transistors

Logic gates

Circuits

Memory

CPU

OS

Browser



LAYERS OF ABSTRACTION

Electrons

Transistors

Logic gates

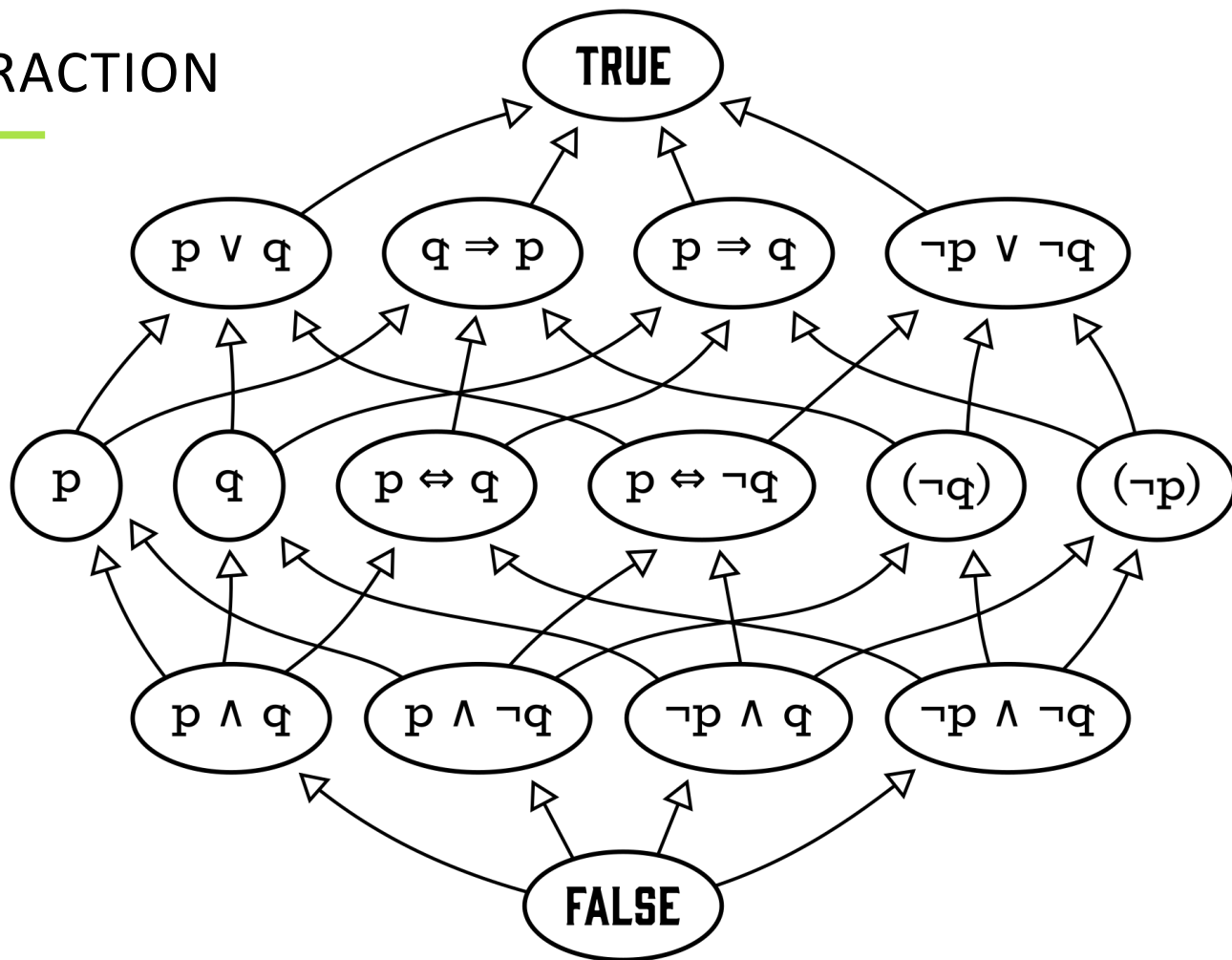
Circuits

Memory

CPU

OS

Browser



LAYERS OF ABSTRACTION

Electrons

Transistors

Logic gates

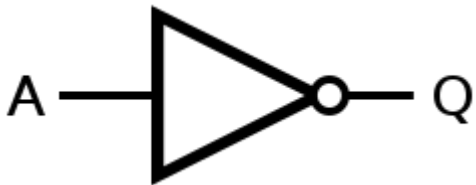
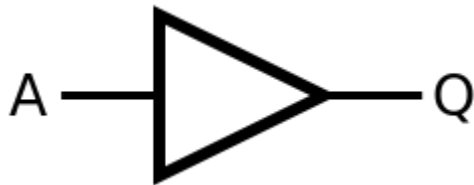
Circuits

Memory

CPU

OS

Browser



LAYERS OF ABSTRACTION

Electrons

Transistors

Logic gates

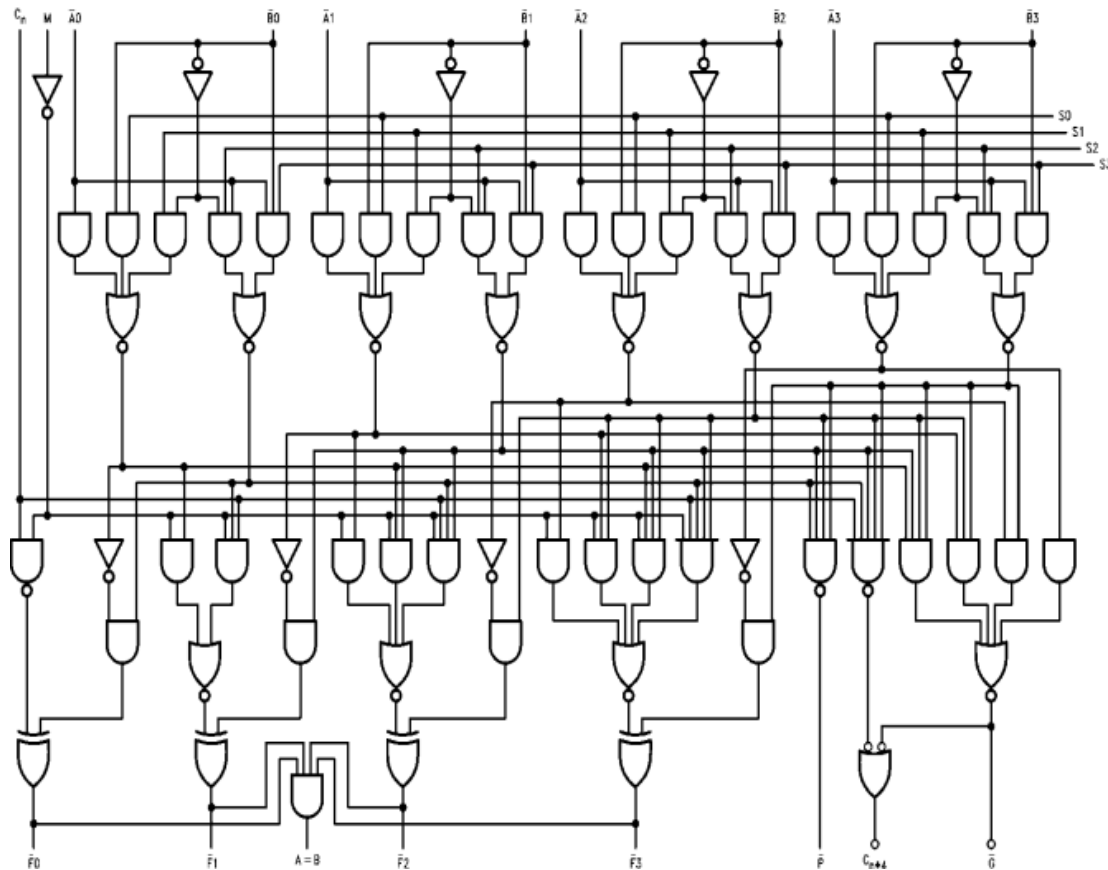
Circuits

Memory

CPU

OS

Browser



LAYERS OF ABSTRACTION

Electrons

Transistors

Logic gates

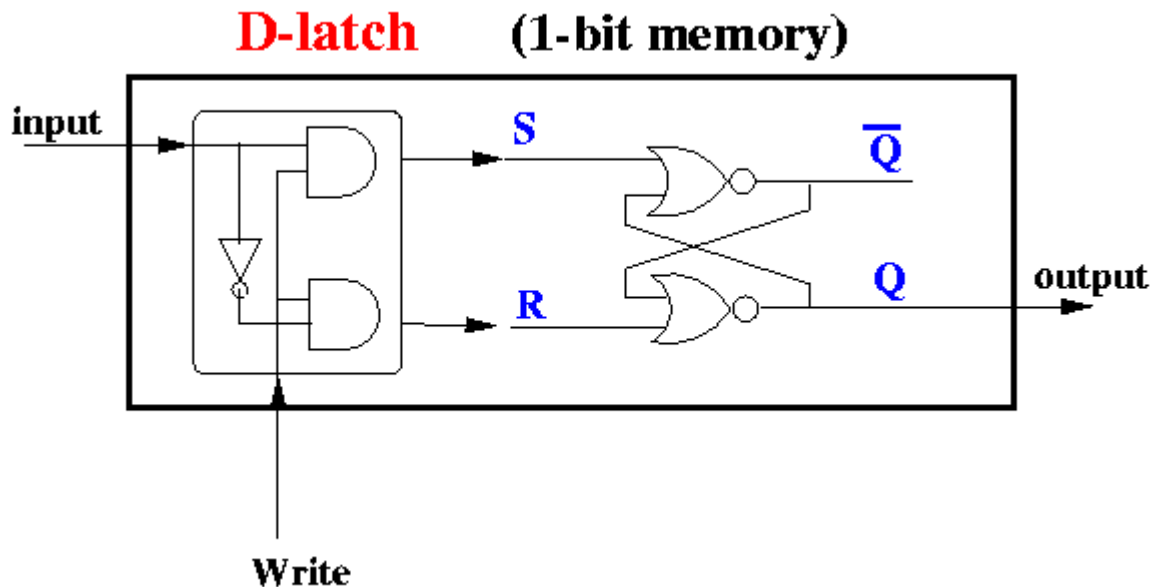
Circuits

Memory

CPU

OS

Browser



LAYERS OF ABSTRACTION

Electrons

Transistors

Logic gates

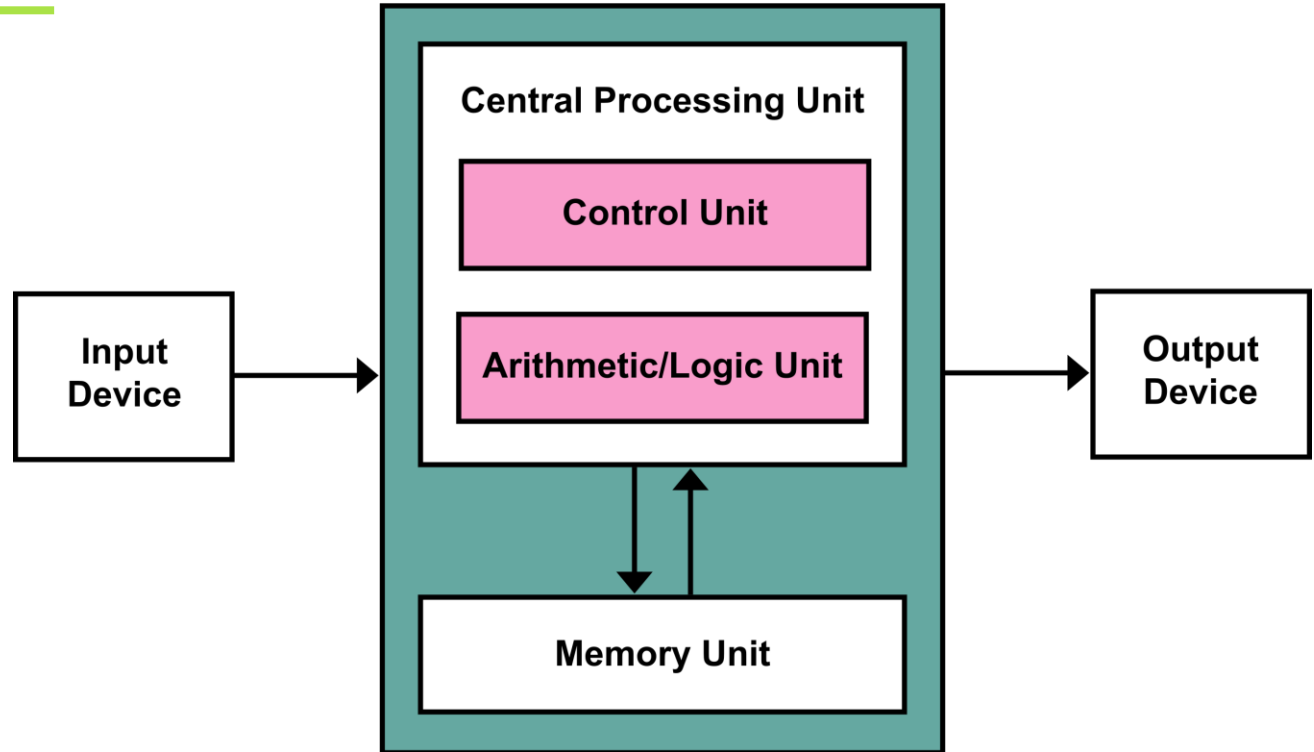
Circuits

Memory

CPU

OS

Browser



LAYERS OF ABSTRACTION

Electrons

Transistors

Logic gates

Circuits

Memory

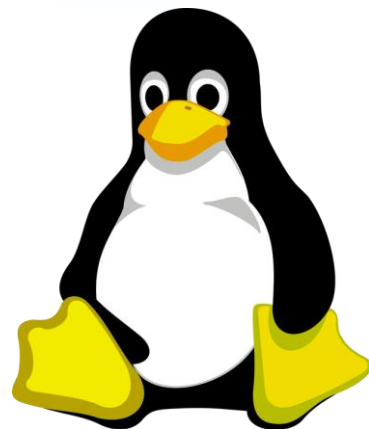
CPU

os

Browser



macOS



LAYERS OF ABSTRACTION

Electrons

Transistors

Logic gates

Circuits

Memory

CPU

OS

Browser



ALL PROBLEMS IN
COMPUTER SCIENCE CAN
BE SOLVED BY ANOTHER
LEVEL OF INDIRECTION,
EXCEPT FOR THE PROBLEM
OF TOO MANY LAYERS OF
INDIRECTION.

SOME LEAKY ABSTRACTIONS

Loop tiling

Network – TCP/IP

SQL

NFS, SMB

PNGs

Spectre/Meltdown

...and more



WHAT IS THE WRONG LEVEL OF
ABSTRACTION?



THE ROAD TO
HELL IS PAVED
WITH GOOD
INTENTIONS



EXAMPLE

Super cool title!

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum rutrum urna varius, placerat ex tempor, dapibus felis. Maecenas eleifend, nulla eu tristique ultrices, dolor eros hendrerit lorem, ac feugiat arcu nibh vel nisi. Vestibulum erat ipsum, volutpat id leo et, dictum euismod felis. Suspendisse dolor mi, eleifend at metus sit amet, vestibulum accumsan arcu. Nullam euismod condimentum libero, sit amet sodales lacus. Etiam mattis dolor felis, id aliquet sem rutrum non. Aenean non aliquet dui. Nam euismod mollis mauris, in venenatis neque viverra et. Praesent ut consectetur dui.

Sed eu felis dolor. Donec sit amet luctus mauris. Aenean blandit rutrum augue in luctus. Aenean sed aliquet risus. Pellentesque sit amet neque eget metus dictum tristique vel leo. Sed cursus laoreet quam, id scelerisque metus cursus vitae. Aliquam ullamcorper justo eu urna tincidunt, et rhoncus elit pellentesque. Donec efficitur, diam sed pharetra elementum, tortor felis ultricies est, ut bibendum nisi enim eget turpis. Nulla sodales vitae ante et tempor. Morbi at felis ac justo laoreet sollicitudin.

Nulla aliquam est quis arcu consectetur eleifend. Duis a rhoncus diam, nec feugiat nisl. Aenean facilisis nunc in ante finibus ultricies. Etiam est risus, pulvinar venenatis magna vitae, dictum porta eros. Pellentesque interdum, massa ut aliquet dapibus, est turpis lobortis justo, eleifend commodo felis libero ut metus. Fusce at accumsan odio. Cras elementum scelerisque tellus eu auctor. Nunc ac vehicula elit. Nunc in tellus nec sem fermentum condimentum. Praesent id tincidunt quam.

OK

```
import * as React from "react";
```

```
export type ModalProps = {  
  title: string  
  content: JSX.Element | JSX.Element[]  
}
```

```
type ModalState = {}
```

```
export class Modal
```

```
  closeModal
```

```
  render
```

```
  re
```

```
    title="Super Cool Title!"  
    content="Lorem Ipsum ..." />
```

```
    <h1>{this.props.title}</h1>
```

```
    <div>{this.props.content}</div>
```

```
    <a onClick={this.closeModal}>
```

```
  </div>
```

```
</>
```

```
}
```

```
}
```

Super cool title!

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum rutrum urna varius, placerat ex tempor, dapibus felis. Maecenas eleifend, nulla eu tristique ultrices, dolor eros hendrerit lorem, ac feugiat arcu nibh vel nisi. Vestibulum erat ipsum, volutpat id leo et dictum euismod felis. Suspendisse dolor mi, eleifend at metus sit amet, vestibulum

OK

```
import * as React from "react";
```

```
export type ModalProps = {  
  title?: string  
  content: JSX.Element | JSX.Element[]  
}
```

```
type ModalState = {}
```

```
export class Modal extends React.Component
```

```
  closeMe = () => { }
```

```
  render() {
```

```
    return <>
```

```
      <div>
```

```
        {this.props.title && <h1>
```

```
        <div>{this.props.content}
```

```
        <a onClick={this.closeMe}
```

```
      </div>
```

```
    </>
```

```
  }
```

```
}
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum rutrum urna varius, placerat ex tempor, dapibus felis. Maecenas eleifend, nulla eu tristique ultrices, dolor eros hendrerit lorem, ac feugiat arcu nibh vel nisi. Vestibulum erat ipsum, volutpat id leo et, dictum euismod felis. Suspendisse dolor mi, eleifend at metus sit amet, vestibulum accumsan arcu. Nullam euismod condimentum libero, sit amet sodales lacus. Etiam mattis dolor felis, id aliquet sem rutrum non. Aenean non aliquet dui. Nam euismod mollis mauris, in venenatis neque viverra et. Praesent ut consectetur dui.

Sed eu felis dolor. Donec sit amet luctus mauris. Aenean blandit rutrum augue in luctus. Aenean sed aliquet risus. Pellentesque sit amet neque eget metus dictum tristique vel leo. Sed cursus laoreet quam, id scelerisque metus cursus vitae. Aliquam ullamcorper justo eu urna tincidunt, et rhoncus elit pellentesque. Donec efficitur, diam sed pharetra elementum, tortor felis ultricies est, ut bibendum nisi enim eget turpis. Nulla sodales vitae ante et tempor. Morbi at felis ac justo laoreet sollicitudin.

Nulla aliquam est quis arcu consectetur eleifend. Duis a rhoncus diam, nec feugiat nisl. Aenean facilisis nunc in ante finibus ultricies. Etiam est risus, pulvinar venenatis magna vitae, dictum porta eros. Pellentesque interdum, massa ut aliquet dapibus, est turpis lobortis justo, eleifend commodo felis libero ut metus. Fusce at accumsan odio. Cras elementum scelerisque tellus eu auctor. Nunc ac vehicula elit. Nunc in tellus nec sem fermentum condimentum. Praesent id tincidunt quam.

OK

```
import * as React from "react";
```

```
export type ModalProps = {  
  title?: string  
  closeInHeader?: boolean
```

Super cool title!

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum rutrum urna varius, placerat ex tempor, dapibus felis. Maecenas eleifend, nulla eu tristique ultrices, dolor eros hendrerit lorem, ac feugiat arcu nibh vel nisi. Vestibulum erat ipsum, volutpat id leo et, dictum euismod felis. Suspendisse dolor mi, eleifend at metus sit amet, vestibulum accumsan arcu. Nullam euismod condimentum libero, sit amet sodales lacus. Etiam mattis dolor felis, id aliquet sem rutrum non. Aenean non aliquet dui. Nam euismod mollis mauris, in venenatis neque viverra et. Praesent ut consectetur dui.

Sed eu felis dolor. Donec sit amet luctus mauris. Aenean blandit rutrum augue in luctus. Aenean sed aliquet risus. Pellentesque sit amet neque eget metus dictum tristique vel vel leo. Sed cursus laoreet quam, id scelerisque metus cursus vitae. Aliquam ullamcorper justo eu urna tincidunt, et rhoncus elit pellentesque. Donec efficitur, diam sed pharetra elementum, tortor felis ultricies est, ut bibendum nisi enim eget turpis. Nulla sodales vitae ante et tempor. Morbi at felis ac justo laoreet sollicitudin.

Nulla aliquam est quis arcu consectetur eleifend. Duis a rhoncus diam, nec feugiat nisl. Aenean facilisis nunc in ante finibus ultricies. Etiam est risus, pulvinar venenatis magna vitae, dictum porta eros. Pellentesque interdum, massa ut aliquet dapibus, est turpis lobortis justo, eleifend commodo felis libero ut metus. Fusce at accumsan odio. Cras elementum scelerisque tellus eu auctor. Nunc ac vehicula elit. Nunc in tellus nec sem fermentum condimentum. Praesent id tincidunt quam.

```
</a>}
```

```
</div>
```

```
</>
```

```
}
```

```
}
```

```
nt
```

```
| st
```

```
nt<Mo
```

```
>
```

```
header
```

```
}</di
```

```
r ==
```

```
Content ? this.props.closeBtnContent : "OK"}
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum rutrum urna varius, placerat ex tempor, dapibus felis. Maecenas eleifend, nulla eu tristique ultrices, dolor eros hendrerit lorem, ac feugiat arcu nibh vel nisi. Vestibulum erat ipsum, volutpat id leo et, dictum euismod felis. Suspendisse dolor mi, eleifend at metus sit amet, vestibulum accumsan arcu. Nullam euismod condimentum libero, sit amet sodales lacus. Etiam mattis dolor felis, id aliquet sem rutrum non. Aenean non aliquet dui. Nam euismod mollis mauris, in venenatis neque viverra et. Praesent ut consectetur dui.

Sed eu felis dolor. Donec sit amet luctus mauris. Aenean blandit rutrum augue in luctus. Aenean sed aliquet risus. Pellentesque sit amet neque eget metus dictum tristique vel vel leo. Sed cursus laoreet quam, id scelerisque metus cursus vitae. Aliquam ullamcorper justo eu urna tincidunt, et rhoncus elit pellentesque. Donec efficitur, diam sed pharetra elementum, tortor felis ultricies est, ut bibendum nisi enim eget turpis. Nulla sodales vitae ante et tempor. Morbi at felis ac justo laoreet sollicitudin.

Nulla aliquam est quis arcu consectetur eleifend. Duis a rhoncus diam, nec feugiat nisl. Aenean facilisis nunc in ante finibus ultricies. Etiam est risus, pulvinar venenatis magna vitae, dictum porta eros. Pellentesque interdum, massa ut aliquet dapibus, est turpis lobortis justo, eleifend commodo felis libero ut metus. Fusce at accumsan odio. Cras elementum scelerisque tellus eu auctor. Nunc ac vehicula elit. Nunc in tellus nec sem fermentum condimentum. Praesent id tincidunt quam.


```
import * as React from "react";
```

```
export type ModalProps = {  
  title?: string  
  fullscreen?: boolean  
  closeInHeader?: boolean  
  scrollingBehavior?: "a" | "b" | "c"  
  closeBtnContent?: string | JSX.Element  
  content: JSX.Element | JSX.Element  
}
```

```
type ModalState = {}
```

```
export class Modal extends React.Component
```

```
  closeMe = () => { }
```

```
  render() {
```

```
    return <>
```

```
      <div>
```

```
        {this.props.title && <
```

```
          {this.props.title}
```

```
          {this.props.closeI
```

```
        </h1>}
```

```
        <div>{this.props.conte
```

```
          {this.props.closeInHea
```

```
          {this.props.closeB
```

```
        </a>}
```

```
      </div>
```

```
    </>
```

```
  }
```

```
}
```

Super cool title!

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum rutrum urna varius, placerat ex tempor, dapibus felis. Maecenas eleifend, nulla eu tristique ultrices, dolor eros hendrerit lorem, ac feugiat arcu nibh vel nisi. Vestibulum erat ipsum, volutpat id leo et, dictum euismod felis. Suspendisse dolor mi, eleifend at metus sit amet, vestibulum accumsan arcu. Nullam euismod condimentum libero, sit amet sodales lacus. Etiam mattis dolor felis, id aliquet sem rutrum non. Aenean non aliquet dui. Nam euismod mollis mauris, in venenatis neque viverra et. Praesent ut consectetur dui.

Sed eu felis dolor. Donec sit amet luctus mauris. Aenean blandit rutrum augue in luctus. Aenean sed aliquet risus. Pellentesque sit amet neque eget metus dictum tristique vel leo. Sed cursus laoreet quam, id scelerisque metus cursus vitae. Aliquam ullamcorper justo eu urna tincidunt, et rhoncus elit pellentesque. Donec efficitur, diam sed pharetra elementum, tortor felis ultricies est, ut bibendum nisi enim eget turpis. Nulla sodales vitae ante et tempor. Morbi at felis ac justo laoreet sollicitudin.

Nulla aliquam est quis arcu consectetur eleifend. Duis a rhoncus diam, nec feugiat nisl. Aenean facilisis nunc in ante finibus ultricies. Etiam est risus, pulvinar venenatis magna vitae, dictum porta eros. Pellentesque interdum, massa ut aliquet dapibus, est turpis lobortis justo, eleifend commodo felis libero ut metus. Fusce at accumsan odio. Cras elementum scelerisque tellus eu auctor. Nunc ac vehicula elit. Nunc in tellus nec sem fermentum condimentum. Praesent id tincidunt quam.

OK



But Wait...
**There's
MORE!**

```
export type ModalProps = {  
  title?: string  
  fullscreen?: boolean  
  closeInHeader?: boolean  
  size?: {  
    w: number  
    h: number  
  }  
  scrollingBehavior?: "overscroll" | "inside" | "outside"  
  closeBtnContent?: string | JSX.Element  
  content?: JSX.Element | JSX.Element[] | string | number  
  position?: {  
    t: number  
    l: number  
    b: number  
    r: number  
  }  
  clickClose?: boolean  
  escapeClose?: boolean  
  closeExisting?: boolean  
  cssClasses?: { /* */ }  
}
```

WRONG LEVELS OF ABSTRACTION

- Wrong problem on the right level
- Right problem on the wrong level
- Wrong

<Modal

```
title="Super Cool Title!"  
content="Lorem Ipsum ..." />
```

```
export type ModalProps = {
  title?: string
  fullscreen?: boolean
  closeInHeader?: boolean
  size?: {
    w: number
    h: number
  }
  scrollingBehavior?: "overscroll" | "inside" | "outside"
  closeBtnContent?: string | JSX.Element
  content?: JSX.Element | JSX.Element[] | string | number
  position?: {
    t: number
    l: number
    b: number
    r: number
  }
  clickClose?: boolean
  escapeClose?: boolean
  closeExisting?: boolean
  cssClasses?: { /* */ }
}
```

WHAT IS THE RIGHT LEVEL OF
ABSTRACTION?

BASIC RULES

Simplify reality

Ignore the extraneous details

Focus on what is important

Divide and conquer

Don't abstract because you probably would someday need to extend "this" with "that"

Super cool title!

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum rutrum urna varius, placerat ex tempor, dapibus felis. Maecenas eleifend, nulla eu tristique ultrices, dolor eros hendrerit lorem, ac feugiat arcu nibh vel nisi. Vestibulum erat ipsum, volutpat id leo et, dictum euismod felis. Suspendisse dolor mi, eleifend at metus sit amet, vestibulum accumsan arcu. Nullam euismod condimentum libero, sit amet sodales lacus. Etiam mattis dolor felis, id aliquet sem rutrum non. Aenean non aliquet dui. Nam euismod mollis mauris, in venenatis neque viverra et. Praesent ut consectetur dui.

Sed eu felis dolor. Donec sit amet luctus mauris. Aenean blandit rutrum augue in luctus. Aenean sed aliquet risus. Pellentesque sit amet neque eget metus dictum tristique vel leo. Sed cursus laoreet quam, id scelerisque metus cursus vitae. Aliquam ullamcorper justo eu urna tincidunt, et rhoncus elit pellentesque. Donec efficitur, diam sed pharetra elementum, tortor felis ultricies est, ut bibendum nisi enim eget turpis. Nulla sodales vitae ante et tempor. Morbi at felis ac justo laoreet sollicitudin.

Nulla aliquam est quis arcu consectetur eleifend. Duis a rhoncus diam, nec feugiat nisi. Aenean facilisis nunc in ante finibus ultricies. Etiam est risus, pulvinar venenatis magna vitae, dictum porta eros. Pellentesque interdum, massa ut aliquet dapibus, est turpis lobortis justo, eleifend commodo felis libero ut metus. Fusce at accumsan odio. Cras elementum scelerisque tellus eu auctor. Nunc ac vehicula elit. Nunc in tellus nec sem fermentum condimentum. Praesent id tincidunt quam.

OK

Super cool title!

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum rutrum urna varius, placerat ex tempor, dapibus felis. Maecenas eleifend, nulla eu tristique ultrices, dolor eros hendrerit lorem, ac feugiat arcu nibh vel nisi. Vestibulum erat ipsum, volutpat id leo et, dictum euismod felis. Suspendisse dolor mi, eleifend at metus sit amet, vestibulum accumsan arcu. Nullam euismod condimentum libero, sit amet sodales lacus. Etiam mattis dolor felis, id aliquet sem rutrum non. Aenean non aliquet dui. Nam euismod mollis mauris, in venenatis neque viverra et. Praesent ut consectetur dui.

Sed eu felis dolor. Donec sit amet luctus mauris. Aenean blandit rutrum augue in luctus. Aenean sed aliquet risus. Pellentesque sit amet neque eget metus dictum tristique vel leo. Sed cursus laoreet quam, id scelerisque metus cursus vitae. Aliquam ullamcorper justo eu urna tincidunt, et rhoncus elit pellentesque. Donec efficitur, diam sed pharetra elementum, tortor felis ultricies est, ut bibendum nisi enim eget turpis. Nulla sodales vitae ante et tempor. Morbi at felis ac justo laoreet sollicitudin.

Nulla aliquam est quis arcu consectetur eleifend. Duis a rhoncus diam, nec feugiat nisi. Aenean facilisis nunc in ante finibus ultricies. Etiam est risus, pulvinar venenatis magna vitae, dictum porta eros. Pellentesque interdum, massa ut aliquet dapibus, est turpis lobortis justo, eleifend commodo felis libero ut metus. Fusce at accumsan odio. Cras elementum scelerisque tellus eu auctor. Nunc ac vehicula elit. Nunc in tellus nec sem fermentum condimentum. Praesent id tincidunt quam.

OK

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum rutrum urna varius, placerat ex tempor, dapibus felis. Maecenas eleifend, nulla eu tristique ultrices, dolor eros hendrerit lorem, ac feugiat arcu nibh vel nisi. Vestibulum erat ipsum, volutpat id leo et, dictum euismod felis. Suspendisse dolor mi, eleifend at metus sit amet, vestibulum accumsan arcu. Nullam euismod condimentum libero, sit amet sodales lacus. Etiam mattis dolor felis, id aliquet sem rutrum non. Aenean non aliquet dui. Nam euismod mollis mauris, in venenatis neque viverra et. Praesent ut consectetur dui.

Sed eu felis dolor. Donec sit amet luctus mauris. Aenean blandit rutrum augue in luctus. Aenean sed aliquet risus. Pellentesque sit amet neque eget metus dictum tristique vel leo. Sed cursus laoreet quam, id scelerisque metus cursus vitae. Aliquam ullamcorper justo eu urna tincidunt, et rhoncus elit pellentesque. Donec efficitur, diam sed pharetra elementum, tortor felis ultricies est, ut bibendum nisi enim eget turpis. Nulla sodales vitae ante et tempor. Morbi at felis ac justo laoreet sollicitudin.

Nulla aliquam est quis arcu consectetur eleifend. Duis a rhoncus diam, nec feugiat nisi. Aenean facilisis nunc in ante finibus ultricies. Etiam est risus, pulvinar venenatis magna vitae, dictum porta eros. Pellentesque interdum, massa ut aliquet dapibus, est turpis lobortis justo, eleifend commodo felis libero ut metus. Fusce at accumsan odio. Cras elementum scelerisque tellus eu auctor. Nunc ac vehicula elit. Nunc in tellus nec sem fermentum condimentum. Praesent id tincidunt quam.

X

Super cool title!

X

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum rutrum urna varius, placerat ex tempor, dapibus felis. Maecenas eleifend, nulla eu tristique ultrices, dolor eros hendrerit lorem, ac feugiat arcu nibh vel nisi. Vestibulum erat ipsum, volutpat id leo et, dictum euismod felis. Suspendisse dolor mi, eleifend at metus sit amet, vestibulum accumsan arcu. Nullam euismod condimentum libero, sit amet sodales lacus. Etiam mattis dolor felis, id aliquet sem rutrum non. Aenean non aliquet dui. Nam euismod mollis mauris, in venenatis neque viverra et. Praesent ut consectetur dui.

Sed eu felis dolor. Donec sit amet luctus mauris. Aenean blandit rutrum augue in luctus. Aenean sed aliquet risus. Pellentesque sit amet neque eget metus dictum tristique vel leo. Sed cursus laoreet quam, id scelerisque metus cursus vitae. Aliquam ullamcorper justo eu urna tincidunt, et rhoncus elit pellentesque. Donec efficitur, diam sed pharetra elementum, tortor felis ultricies est, ut bibendum nisi enim eget turpis. Nulla sodales vitae ante et tempor. Morbi at felis ac justo laoreet sollicitudin.

Nulla aliquam est quis arcu consectetur eleifend. Duis a rhoncus diam, nec feugiat nisi. Aenean facilisis nunc in ante finibus ultricies. Etiam est risus, pulvinar venenatis magna vitae, dictum porta eros. Pellentesque interdum, massa ut aliquet dapibus, est turpis lobortis justo, eleifend commodo felis libero ut metus. Fusce at accumsan odio. Cras elementum scelerisque tellus eu auctor. Nunc ac vehicula elit. Nunc in tellus nec sem fermentum condimentum. Praesent id tincidunt quam.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum rutrum urna varius, placerat ex tempor, dapibus felis. Maecenas eleifend, nulla eu tristique ultrices, dolor eros hendrerit lorem, ac feugiat arcu nibh vel nisi. Vestibulum erat ipsum, volutpat id leo et, dictum euismod felis. Suspendisse dolor mi, eleifend at metus sit amet, vestibulum accumsan arcu. Nullam euismod condimentum libero, sit amet sodales lacus. Etiam mattis dolor felis, id aliquet sem rutrum non. Aenean non aliquet dui. Nam euismod mollis mauris, in venenatis neque viverra et. Praesent ut consectetur dui.

Sed eu felis dolor. Donec sit amet luctus mauris. Aenean blandit rutrum augue in luctus. Aenean sed aliquet risus. Pellentesque sit amet neque eget metus dictum tristique vel leo. Sed cursus laoreet quam, id scelerisque metus cursus vitae. Aliquam ullamcorper justo eu urna tincidunt, et rhoncus elit pellentesque. Donec efficitur, diam sed pharetra elementum, tortor felis ultricies est, ut bibendum nisi enim eget turpis. Nulla sodales vitae ante et tempor. Morbi at felis ac justo laoreet sollicitudin.

Nulla aliquam est quis arcu consectetur eleifend. Duis a rhoncus diam, nec feugiat nisi. Aenean facilisis nunc in ante finibus ultricies. Etiam est risus, pulvinar venenatis magna vitae, dictum porta eros. Pellentesque interdum, massa ut aliquet dapibus, est turpis lobortis justo, eleifend commodo felis libero ut metus. Fusce at accumsan odio. Cras elementum scelerisque tellus eu auctor. Nunc ac vehicula elit. Nunc in tellus nec sem fermentum condimentum. Praesent id tincidunt quam.

OK


```
import * as React from "react";
```

```
export type DialogProps = {  
  open: boolean  
  onClose(): void  
}
```

```
export class Dialog extends React.PureComponent<DialogProps>{  
  onClickOutsideOfDialog() { this.props.onClose(); }  
  /* event listeners and etc. */  
  render() {  
    return <>  
      {this.props.open && <div>{this.props.children}</div>}  
    </>  
  }  
}
```

```
import React from "react";
import { Dialog } from "../dialog";

export function AlertDialog({ isOpen, handleAccept, handleDecline, content }) {
  const [open, setOpen] = React.useState(false);
  const handleClose = () => { setOpen(false); };

  setOpen(isOpen);

  return <Dialog open={open} onClose={handleClose}>
    <h1>Alert!</h1>
    {content}
    <button onClick={handleAccept}>Disagree</button>
    <button onClick={handleDecline}>Agree</button>
  </Dialog>;
}
```

```
import React from "react";
import { Dialog } from "../dialog";

export function NotificationDialog({ isOpen, content }) {
  const [open, setOpen] = React.useState(false);
  const handleClose = () => { setOpen(false); };

  setOpen(isOpen);

  return <Dialog open={open} onClose={handleClose}>
    {content}
  </Dialog>;
}
```

```
import React from "react";
import { Logger } from "logger";
import { Dialog } from "../dialog";

export class TrackingDialog extends Dialog {
  onClickOutsideOfDialog() {
    Logger.log("User closed the popup...")
    super.onClickOutsideOfDialog()
  }
}
```

PRACTICAL ADVICE

YAGNI / KISS

DRY

DDIY

YAGNI / KISS

Less design => more flexible code

Assumptions !== Requirements

Don't abstract before you need to!

```
import * as React from "react";

export type ModalProps = {
  title?: string
  fullscreen?: boolean
  closeInHeader?: boolean
  scrollingBehavior?: "a" | "b" | "c"
  closeBtnContent?: string | JSX.Element
  content: JSX.Element | JSX.Element[] | string | number
}
type ModalState = {}

export class Modal extends React.Component<ModalProps, ModalState>{
  closeMe = () => { }
  render() {
    return <>
      <div>
        {this.props.title && <h1>
          {this.props.title}
          {this.props.closeInHeader && <a onClick={this.closeMe}>X</a>}
        </h1>}
        <div>{this.props.content}</div>
        {this.props.closeInHeader === false && <a onClick={this.closeMe}>
          {this.props.closeBtnContent ? this.props.closeBtnContent : "OK"}
        </a>}
      </div>
    </>
  }
}
```

DRY

If you repeat something – make it a function

SoC

```
import React from "react";
import { Dialog } from "../dialog";

export function AlertDialog({ isOpen, handleAccept, handleDecline, content }) {
  const [open, setOpen] = React.useState(false);
  const handleClose = () => { setOpen(false); };

  setOpen(isOpen);

  return <Dialog open={open} onClose={handleClose}>
    <h1>Alert!</h1>
    {content}
    <button onClick={handleAccept}>Disagree</button>
    <button onClick={handleDecline}>Agree</button>
  </Dialog>;
}
```

```
import React from "react";
import { Dialog } from "../dialog";

export function NotificationDialog({ isOpen, content }) {
  const [open, setOpen] = React.useState(false);
  const handleClose = () => { setOpen(false); };

  setOpen(isOpen);

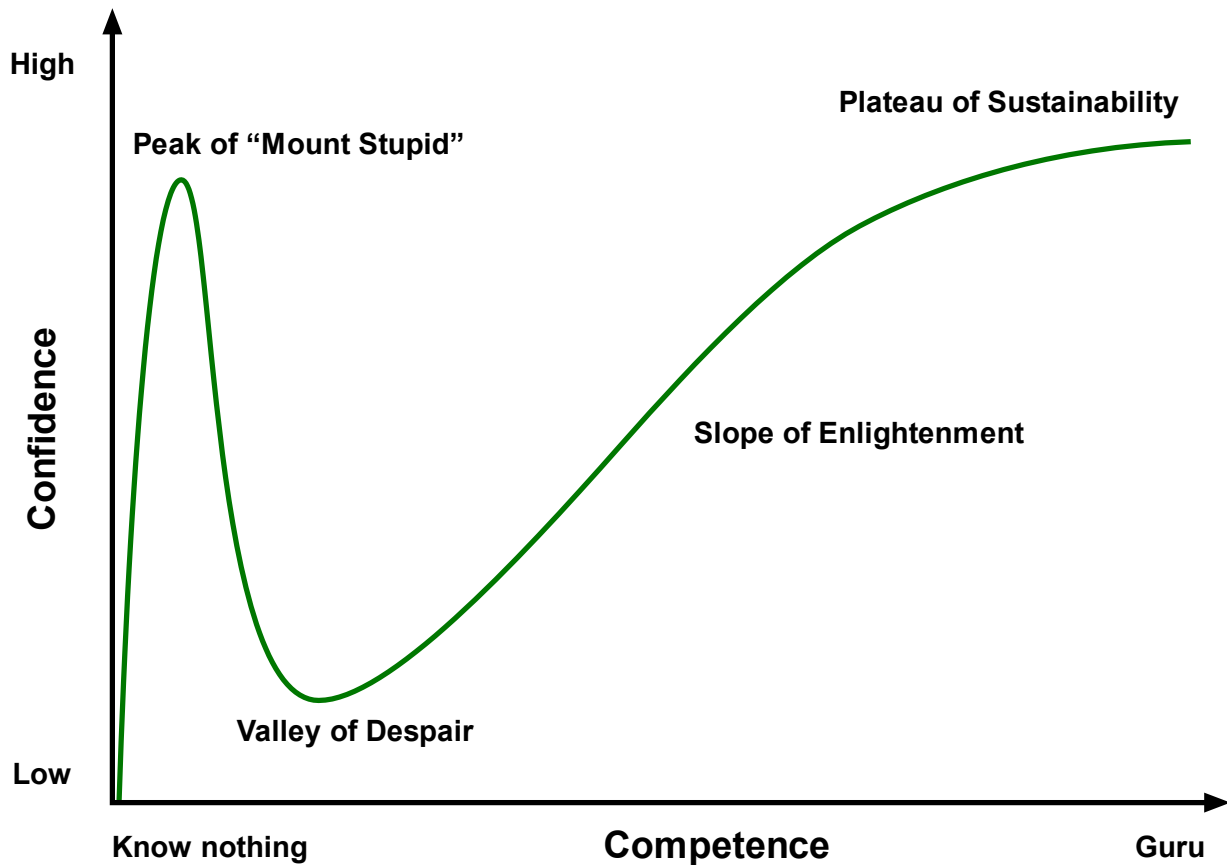
  return <Dialog open={open} onClose={handleClose}>
    {content}
  </Dialog>;
}
```

DDIY

!Pioneers

Dunning Kruger effect

Dunning–Kruger Effect



DDIY

!Pioneers

Dunning Kruger effect

Core business vs Tooling

Dialog React component - Mate

material-ui.com/components/dialogs/

Dialog

Snackbar

Backdrop

Data Display

Avatar

Badge

Chip

Divider

Icons

Material Icons

List

Table

Tooltip

Typography

Utils

Click Away Listener

CSS Baseline

Modal

No SSR

Popover

Popper

Portal

Textarea Autocomplete

Transitions

useMediaQuery

Lab

Search...

ENGLISH

Simple Dialogs

Simple dialogs can provide additional details or actions about a list item. For example, they can display avatars, icons, clarifying subtext, or orthogonal actions (such as adding an account).

Touch mechanics:

Choosing an option immediately commits the option and closes the menu

Touching outside of the dialog, or pressing Back, cancels the action and closes the dialog

Selected: user02@gmail.com

OPEN SIMPLE DIALOG

JS

TS

```
import React from 'react';
import { makeStyles } from '@material-ui/core/styles';
import Button from '@material-ui/core/Button';
import Avatar from '@material-ui/core/Avatar';
import List from '@material-ui/core/List';
import ListItem from '@material-ui/core/ListItem';
import ListItemAvatar from '@material-ui/core/ListItemAvatar';
import ListItemText from '@material-ui/core/ListItemText';
import DialogTitle from '@material-ui/core/DialogTitle';
import Dialog from '@material-ui/core/Dialog';
import PersonIcon from '@material-ui/icons/Person';
import AddIcon from '@material-ui/icons/Add';
import Typography from '@material-ui/core/Typography';
import { blue } from '@material-ui/core/colors';

const emails = ['username@gmail.com', 'user02@gmail.com'];
const useStyles = makeStyles({
  avatar: {
    backgroundColor: blue[100],
    color: blue[600],
  },
});
```

Contents

Simple Dialogs

Alerts

Transitions

Form dialogs

Customized dialogs

Full-screen dialogs

Optional sizes

Responsive full-screen

Confirmation dialogs

Draggable dialog

Scrolling long content

Limitations

Accessibility

API

BONUS: RTFM

Language reference

Framework reference

Host environment

SUMMARY

What is an abstraction?

What happens when abstractions fail?

What is the wrong level of abstraction?

What is the right level of abstraction?

ONE LAST THING...

MASTERY OF ABSTRACTION DEPENDS ON THE ABILITY TO ARTICULATE IDEAS!

CONNECT 😊

LinkedIn: <https://goo.gl/CRRJ2Y>

Mail: m.chaov@draftkings.com

Github: <https://github.com/mchaov>

@raziel_pld