

CS50 for MBAs

Software Engineering

Build

Test

Deploy

Launch

Build

Test

Deploy

Launch

Problem with shared files

```
function showProfile(user) {  
    showProfilePicture(user);  
    showName(user);  
}
```

```
function showName(user) {  
    name = user.getName();  
  
    // [TODO]Task1234 Format Name  
}
```

```
function showProfile(user) {  
    showProfilePicture(user);  
    showFriends(user);  
}
```

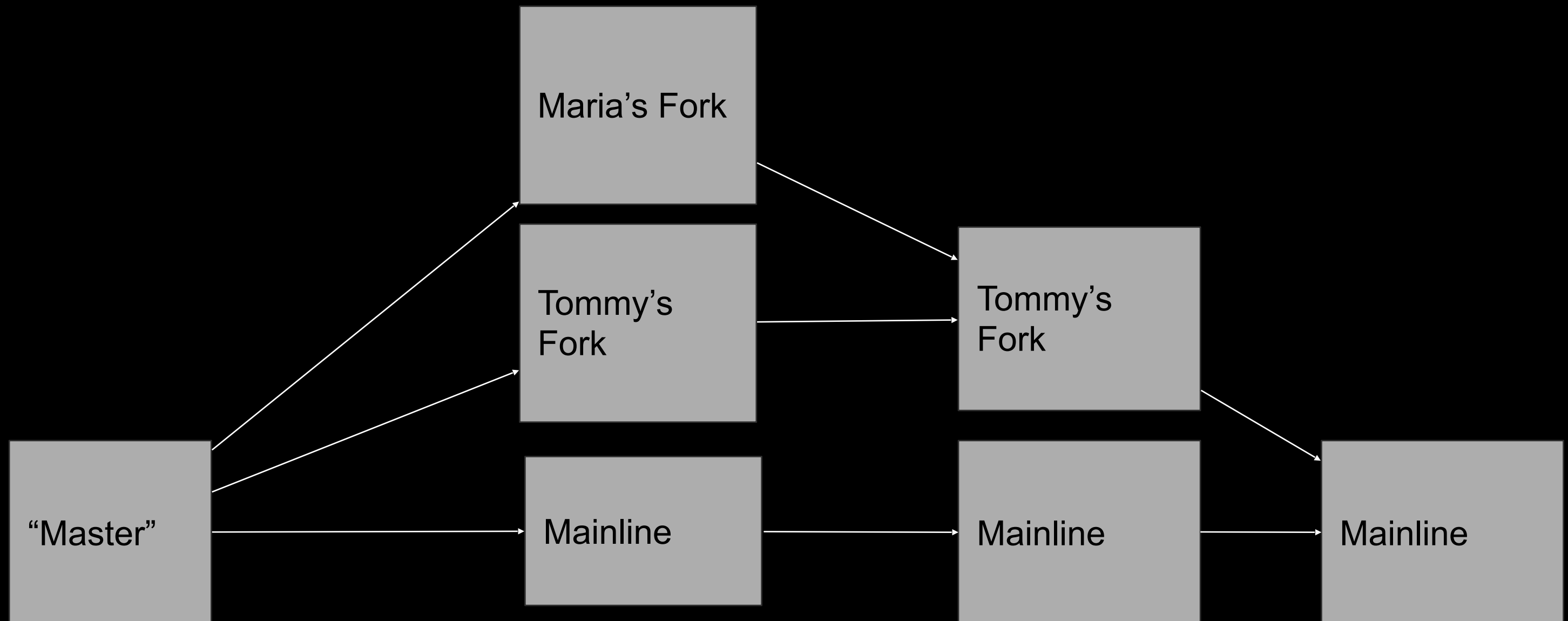
```
function showFriends(user) {  
    friends = user.getFriends();  
  
    // [TODO]Task567 show friends!  
}
```

Maria's job	Tommy's job
-------------	-------------

Version Control

Version Control

git, mercurial



Approaches to version control

Feature Branching

Each feature is worked on in isolation until it's ready to get "merged" back into the main-line (e.g. after say 2 weeks or even 4 months).

Continuous Integration

Engineers commit their code directly to the single main-line codebase (or update it every night).

Approaches to codebase architecture

	Companies	Benefits
Monolith <i>fbsource</i> at Facebook	Google Facebook	Faster code modifications and common tools Integrated teams and company culture
Microservices “Tincup” Uber’s currency rates service	Uber Netflix Ebay	Independent Teams Service Isolation Scalability Easier to open source

Code Review

Improving code by having the problem looked at from different perspectives

src/applications/people/controller/PhabricatorPeopleProfileController.php View Options ▾

Show First 20 Lines • Show All 58 Lines • ▾ Show 20 Lines

```
59     id(new PhabricatorActionView())
60     ->setIcon('image')
61     ->setName(pht('Edit Profile Picture'))
62     ->setHref($this->getApplicationURI('picture/'.$user-
>getID().'/''))
63     ->setDisabled(!$scan_edit)
64     ->setWorkflow(!$scan_edit));
65
66     if ($viewer->getIsAdmin()) {
67
68         $actions->addAction(
69             id(new PhabricatorActionView())
70             ->setIcon('wrench'))
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

```
58     $actions->addAction(
59         id(new PhabricatorActionView())
60         ->setIcon('image')
61         ->setName(pht('Edit Profile Picture'))
62         ->setHref($this->getApplicationURI('picture/'.$user-
>getID().'/''))
63         ->setDisabled(!$scan_edit)
64         ->setWorkflow(!$scan_edit));
65
66     if ($viewer->getIsAdmin()) {
67         $dd = phutil_tag('span', array('class' => 'action-view-caret',
''));
68         $admin_text = pht('Administrate User %s', $dd);
69         $actions->addAction(
70             id(new PhabricatorActionView())
71             ->setIsContainer(true)
72             ->setIcon('user')
73             ->setGroupKey('user.admin'))
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

New Inline Comment

B I T | **☞** | **☰** | **☷** | **</>** | **📅** | **🔒** | **😊** | **🌐**

is GroupKey here a new option?

```
paths = ['/marketplace', '/watch', '/events', '/groups']
domain = 'facebook.com'

urls = []
for path in paths:
    url = 'https://'
    url += domain
    url += path
    urls.append(url)
```

```
paths = ['/marketplace', '/watch', '/events', '/groups']  
domain = 'facebook.com'
```

```
urls = ['https://' + domain + path for path in paths]
```

Engineering Quality





Approaches that help increase engineering productivity & satisfaction

Engineering Quality

Modern codebases

Making codebases more healthy,
modern and coherent

i.e. cleaning “dead” code,
adding type coverage for Python

Empowered teams

Creating a culture that
recognizes solving local
engineering issues that slow
teams down

i.e. testing, planning for better
engineering

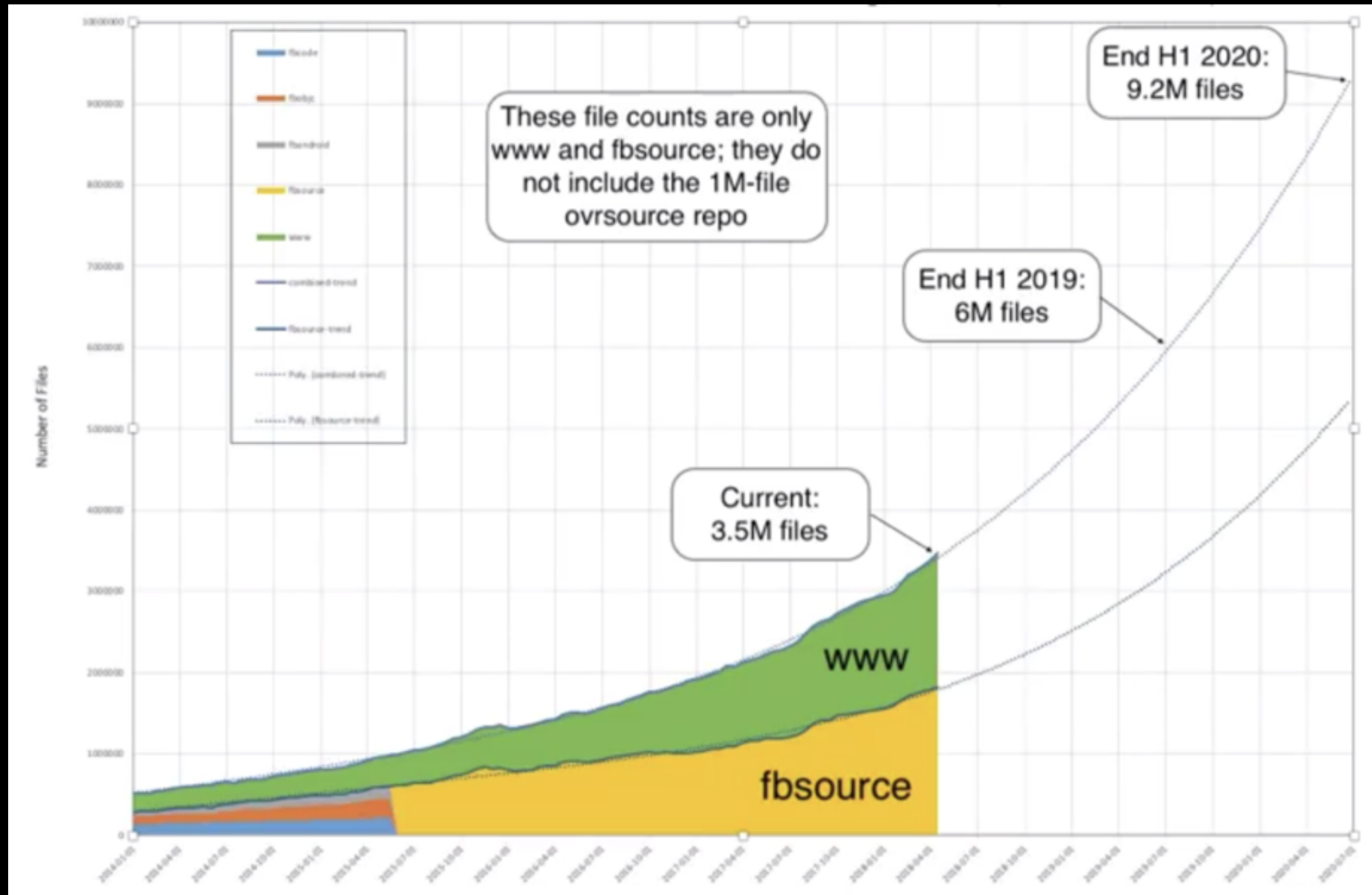
Engineering Productivity Tools

Building a developer experience
that engineers can be proud of

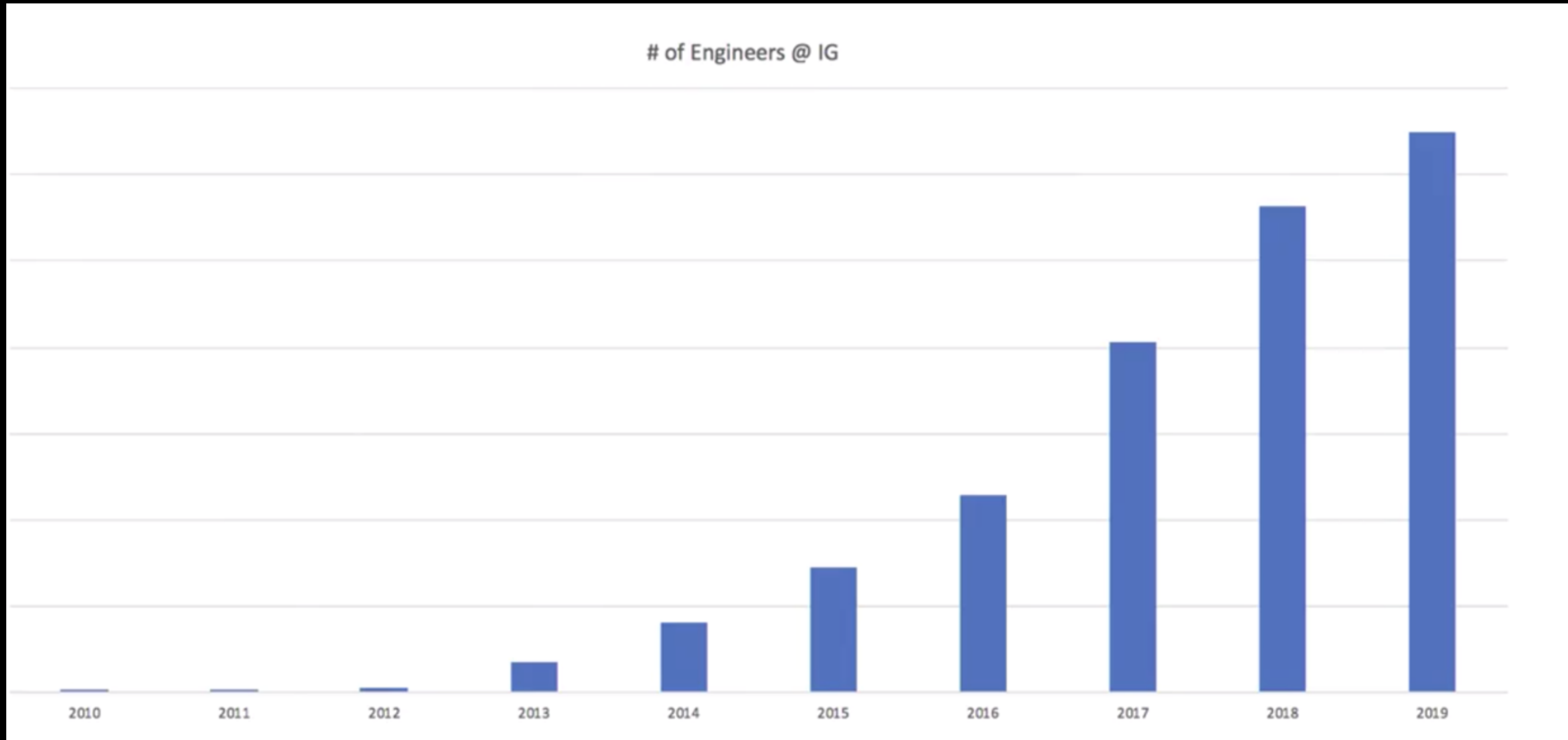
i.e. tools that address the pains
of development

Why do we need an emphasis on engineering quality?

Size of codebase



Size of engineering team



Why does better engineering matter?

Complexity increases over time

Size of codebase x # of engineers =



When scaling, it gets harder to:

- Prioritize code quality improvements
- Build tools for engineering efficiency
- Invest in code maintainability and testability

Which means:

- Teams aren't able to ship impact as fast
- Engineers will be less proud of their codebase
- Slower building of product

Friendster Beta

The new way to meet people

Friendster is an online community that connects people through networks of friends for dating or making new friends.

You can use Friendster to:

- Meet new people to date, through your friends and their friends
- Make new friends
- Help your friends meet new people

Create your own personal and private community, where you can interact with people who are connected to you through networks of mutual friends. It's easy and fun!

[\[Take a Tour | Testimonials | More Info\]](#)



[Sign Up](#)

[Log In](#)

Friendster in 2004, as captured by the Internet Archive.

Friendster

"The problem," [Jonathan Abrams] says, "was that Friendster was having a lot of technology problems." Friendster had raised an additional \$13 million in funding in 2003 and, according to Abrams, **investors weren't focused on patching up the service properly.**

"The fact that we didn't launch those products was a problem, but even more fundamentally, people could **barely log into the website for two years,**" he says. "By the time Facebook and MySpace was doing those things, Friendster had lost a lot of market share in the U.S. for **stability issues.**"



A clean, modern white desk with a white chair, a lamp, a plant, and a window in the background. The desk is positioned in front of a window with a white frame. On the desk, there is a white lamp, a small potted plant in a woven basket, a glass pen holder, a wooden tissue box, and some papers. The chair is white with a black metal base and wooden legs. The background features a white wall with a radiator and a window with a white frame. The overall aesthetic is bright and minimalist.

Better Engineering

Mobile Case Study

Web Apps

Native Apps

Hybrid Apps

Web Apps

Native Apps

Hybrid Apps



HARVARD BUSINESS SCHOOL

About

Academic Programs

Faculty & Research

Alumni

Search





HARVARD BUSINESS SCHOOL

About

Academic Programs

Faculty & Research

Alumni

Media Queries

```
@media screen and (min-width: 920px) {  
    ...  
}
```

```
@media  
(max-width: 919px) and  
(min-width: 650px) {  
    ...  
}
```




 **HARVARD
BUSINESS SCHOOL**

About

Academic Programs

Faculty & Research

Alumni

Search 

Elements Console Sources Network Performance

```
<div class="fullHeader">
  <div class="offNav">...</div>
  <div class="topHead-btn pause-on-hover">...</div>
  <div class="homeContent">
    <video class="vidBox mobile-hidden" muted="muted" autoplay="autoplay" playsinline="playsinline" tabindex="-1" poster="/PublishingImages/fullhd-thumb.jpg">...</video>
    <div class="mobileImg desktop-hidden tablet-hidden"></div>
    <header aria-label="Site Header">
      <div class="fadeBox" style="transform: translateY(0px);">
        <div class="blackBox">
          <h1 class="h1-header">Harvard Business School</h1>
          
          <nav aria-label="Site Utility Navigation">...</nav>
          <nav aria-label="Site Utility Navigation">...</nav>
          <nav aria-label="Site Utility Navigation">...</nav>
          <nav aria-label="Site Utility Navigation">...</nav>
        </div>
        <div class="searchWrapper">...</div>
        <div id="mobileMenu" class="mobileMenu-btn-wrapper">...</div>
        <div class="mobile-offNav desktop-hidden tablet-hidden">...</div>
      </div>
    </header>
  </div>
  <div class="stickHead mobile-hidden" tabindex="-1">...</div>
</div>
```

html body #container div div #skipto-main div div header div div img

Styles Event Listeners DOM Breakpoints Properties Accessibility

Filter :hov .cls +

```
element.style {
}
@media (max-width: 650px)
.blackBox img {
  width: 87%;
  padding-left: 6.5%;
  padding-top: 18px;
}
.blackBox img {
  width: 90%;
  padding-left: 23px;
  padding-top: 18px;
}
@media screen and (max-width: 919px)
img, canvas, iframe, svg {
  max-width: 100%;
}
```

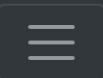
margin -
border -
padding 18
24.375 326.250 x 115.141

Filter Show all

- border-bottom-col... rgb(...)
- border-bottom-sty... none
- border-bottom-wid... 0px
- border-image-outs... 0px
- border-image-repe... stretch

CSS Frameworks

- Bootstrap
- Foundation
- Material UI
- Semantic UI
- ...

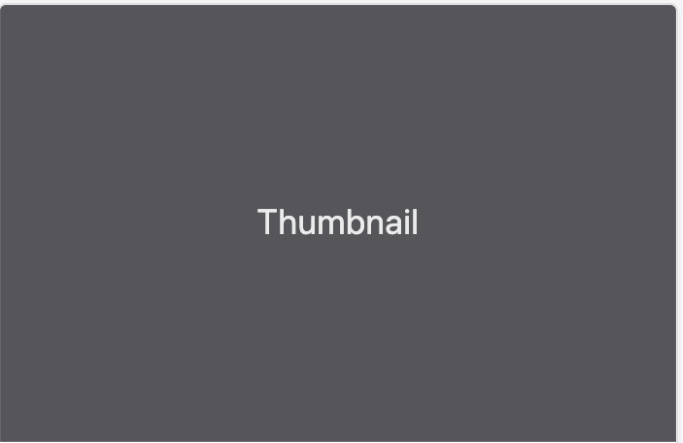


Album example

Something short and leading about the collection below—its contents, the creator, etc. Make it short and sweet, but not too short so folks don't simply skip over it entirely.

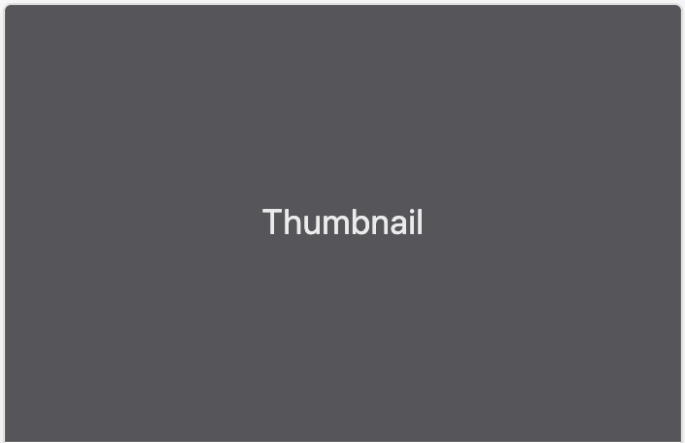
Main call to action

Secondary action



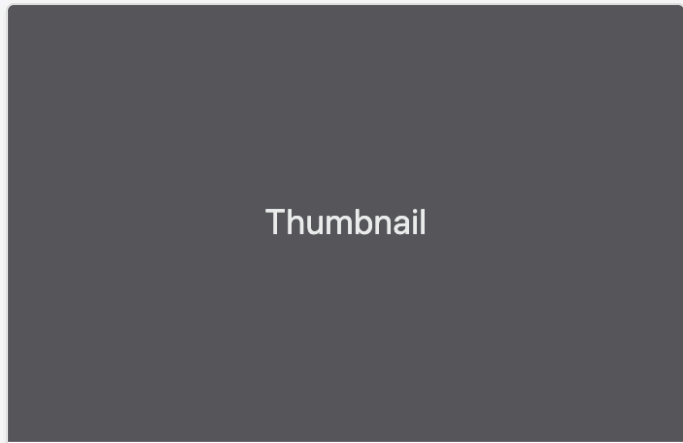
Thumbnail

This is a wider card with supporting text below as a natural lead-in to additional



Thumbnail

This is a wider card with supporting text below as a natural lead-in to additional



Thumbnail

This is a wider card with supporting text below as a natural lead-in to additional

User Agents

Mozilla/5.0 (Macintosh; Intel
Mac OS X 10_14_3) AppleWebKit/
537.36 (KHTML, like Gecko)
Chrome/72.0.3626.119 Safari/
537.36

Mozilla/5.0 (Macintosh; Intel
Mac OS X 10_14_3) AppleWebKit/
537.36 (KHTML, like Gecko)
Chrome/72.0.3626.119 Safari/
537.36

Mozilla/5.0 (iPhone; CPU iPhone
OS 12_1 like Mac OS X)
AppleWebKit/605.1.15 (KHTML,
like Gecko) Version/12.0
Mobile/15E148 Safari/604.1

Mozilla/5.0 (iPhone; CPU iPhone
OS 12_1 like Mac OS X)
AppleWebKit/605.1.15 (KHTML,
like Gecko) Version/12.0
Mobile/15E148 Safari/604.1

```
if 'iphone' in user_agent:  
    ...  
else:  
    ...
```

Tradeoffs: Web Apps

Tradeoffs: Web Apps

+ write code once

+ faster iteration

+ lower learning curve

- slower performance

- limited access to device APIs

- no App/Play Store, home screen icon

Web Apps

Native Apps

Hybrid Apps

Native Platforms

- Android
- BlackBerry
- iOS
- Windows
- ...

Native Platforms

- Android (Java/Kotlin)
- BlackBerry (Java)
- iOS (Objective-C/Swift)
- Windows (C#)
- ...

iOS

```
override func viewDidLoad() {
    let button = UIButton(frame:
        CGRect(x: 0, y: 0, width: 100, height: 50)
    )
    button.setTitle("Hello", for: .normal)
    view.addSubview(button)
    // ...
}
```

Android

```
@Override
protected void onCreate(Bundle state) {
    Button button = new Button(this);
    button.setText("Hello");
    layout.addView(button);
    // ...
}
```

Usage

99.6 percent of new smartphones run Android or iOS

While BlackBerry's market share is a rounding error

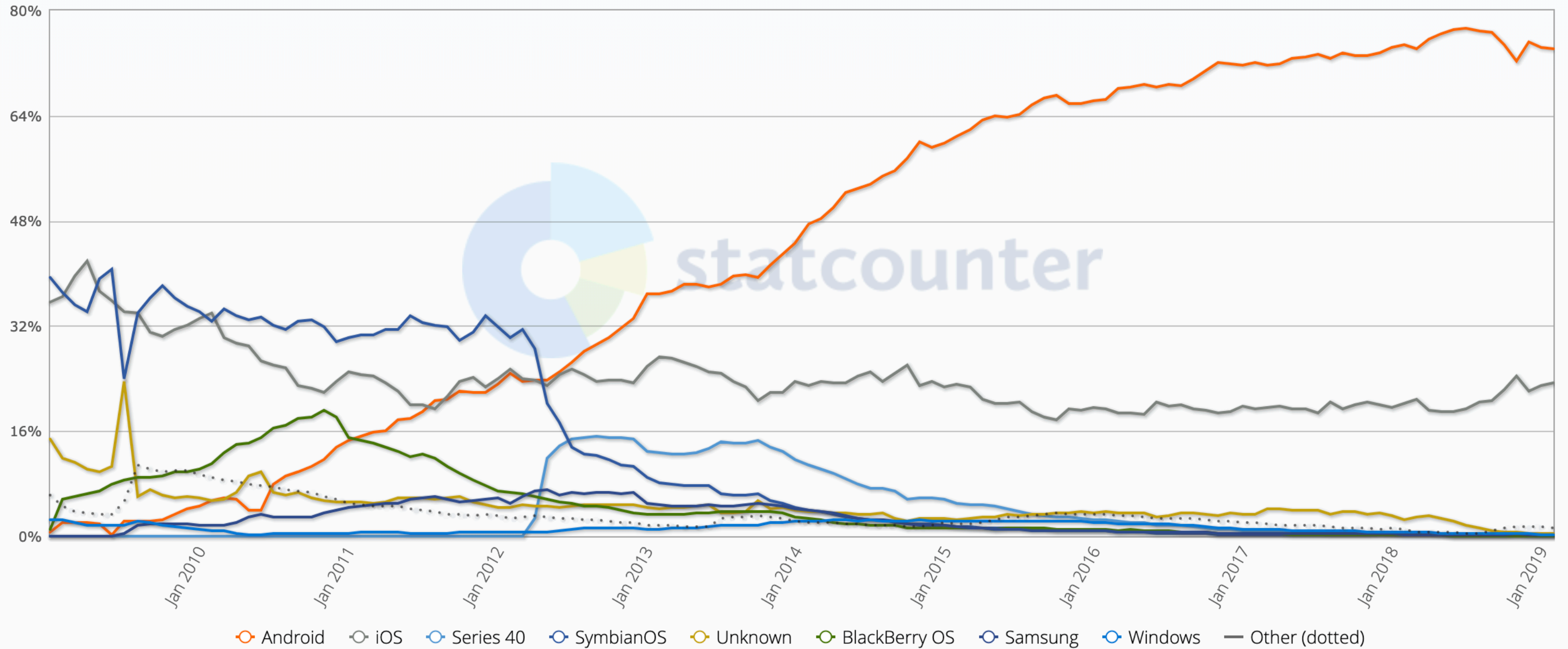
by [James Vincent](#) | [@jjvincent](#) | Feb 16, 2017, 6:11am EST

US

Global

Mobile Operating System Market Share Worldwide

Jan 2009 - Feb 2019



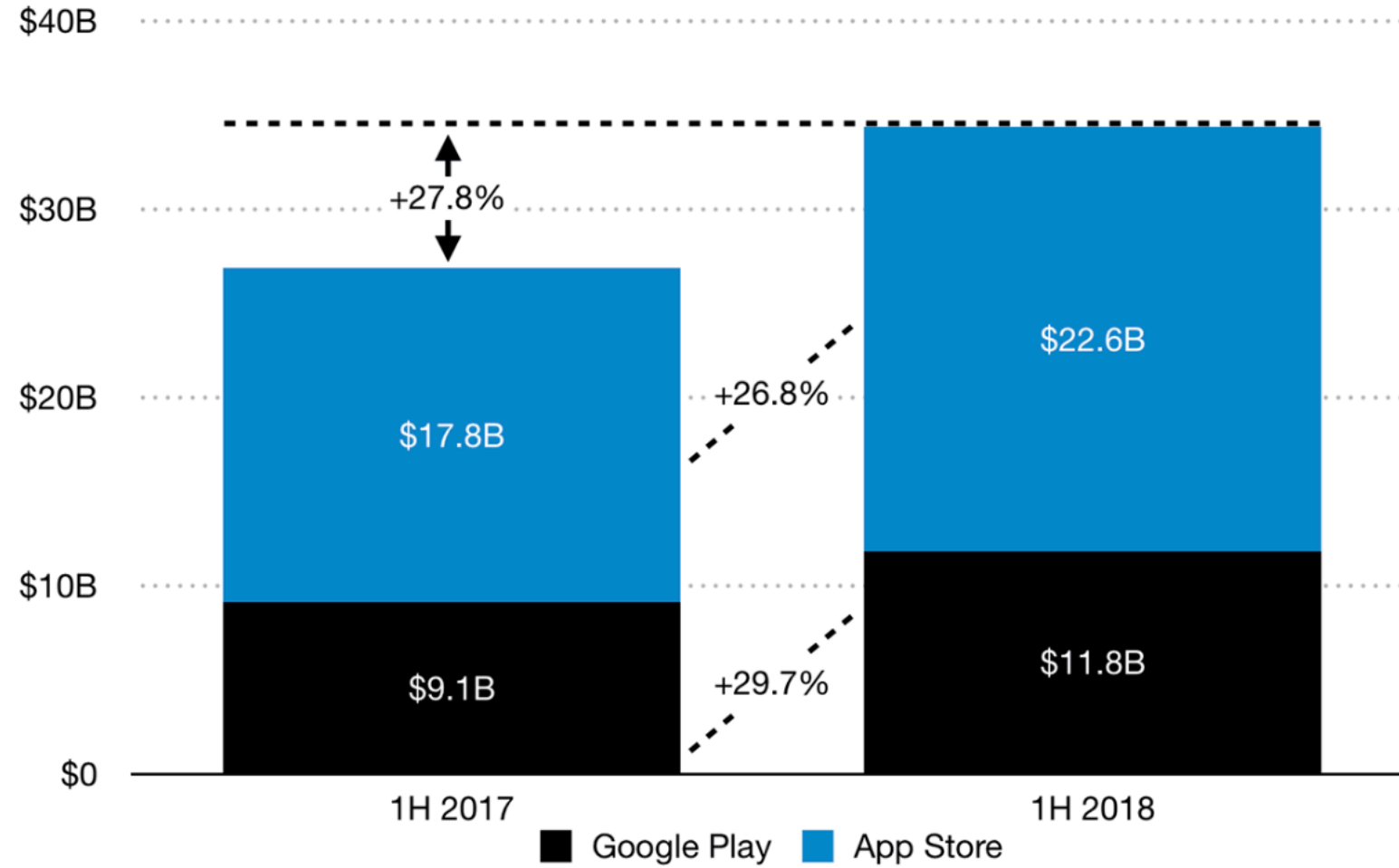
Revenue

Apple's App Store revenue nearly double that of Google Play in first half of 2018

Sarah Perez @sarahintampa / 8 months ago

 Comment

Worldwide Gross App Revenue - First Half 2018



Tradeoffs: Native Apps

Tradeoffs: Native Apps

- + fast performance
- + no limitations on device APIs
- + higher learning curve
- new language per platform
- slower iteration speed
- APIs constantly changing

Web Apps

Native Apps

Hybrid Apps

Web View

Hybrid Frameworks

- Cordova
- React Native
- Titanium
- ...

React

```
class Board extends React.Component {  
  renderSquare(i) {  
    return <Square value={i} />;  
  }  
}
```

```
class Square extends React.Component {  
  ...  
}
```

React Native

```
<SegmentedControlIOS
  values={['One', 'Two']}
  selectedIndex={this.state.selectedIndex}
  onChange={(event) => {
    this.setState({
      selectedIndex:
event.nativeEvent.selectedSegmentIndex
    });
  }}
/>
```

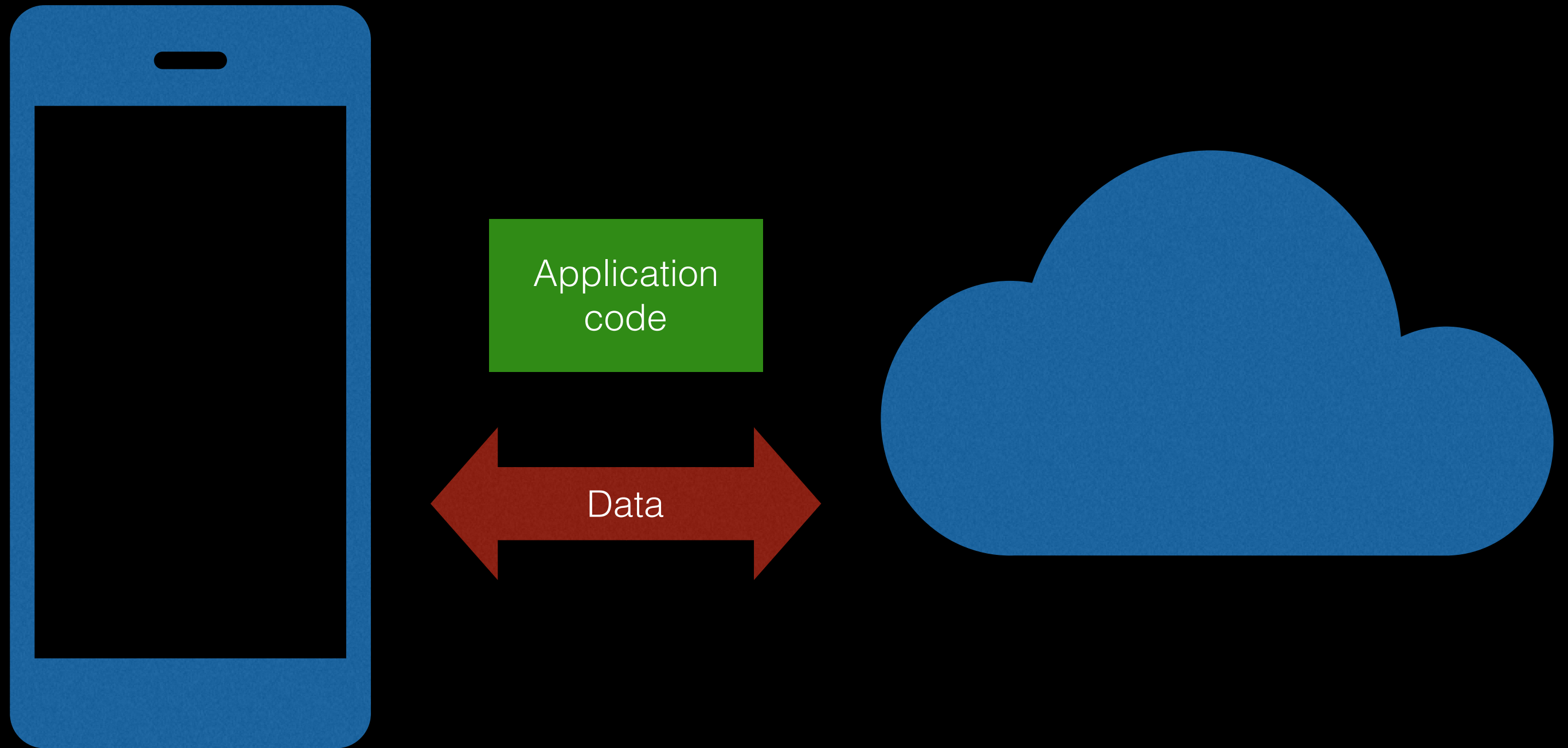
Tradeoffs: Hybrid Apps

Tradeoffs: Hybrid Apps

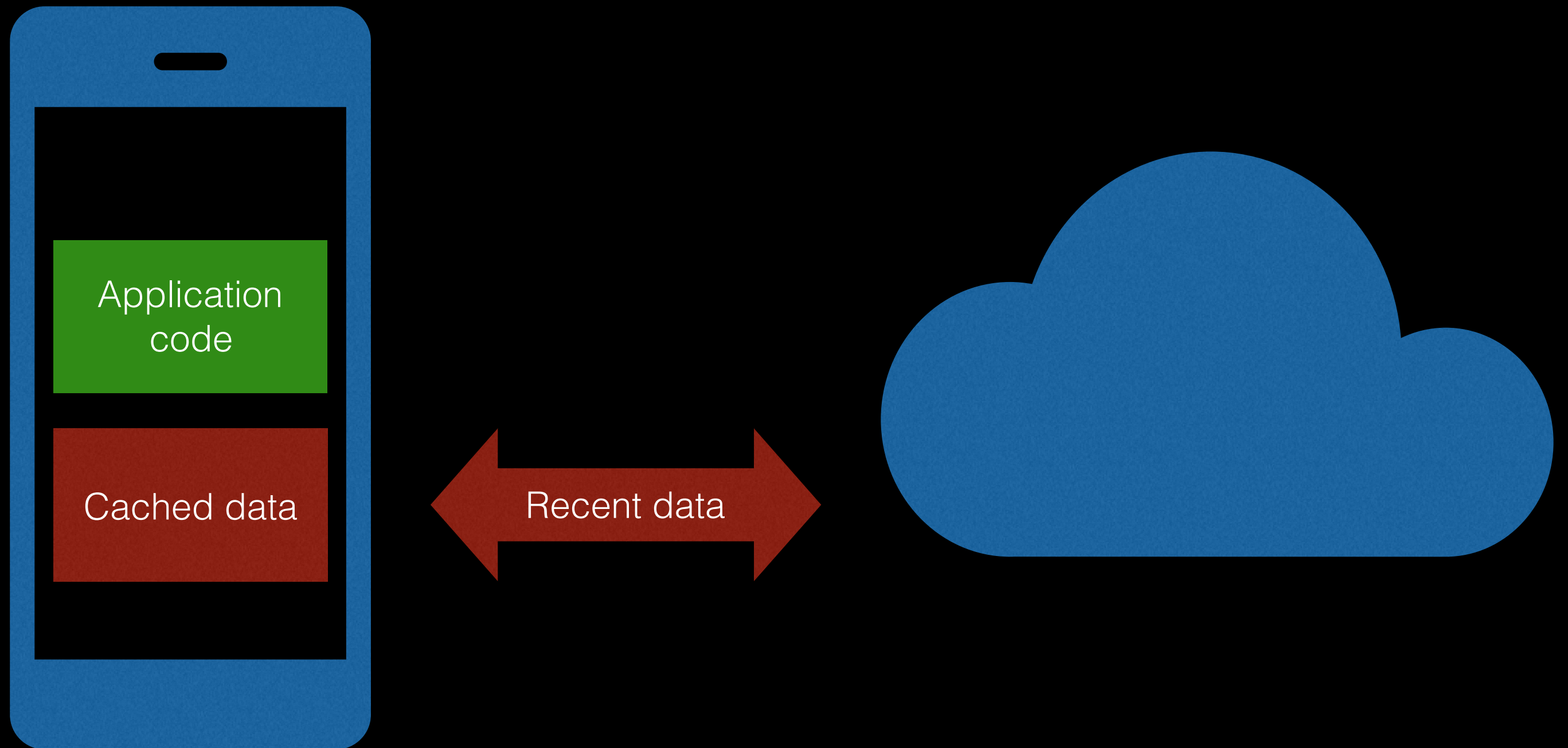
- + fast iteration speed
- + write code mostly once
- + reasonable performance
- rely on frameworks + webviews
- lower performance than native
- some limits on API access

Comparing Approaches

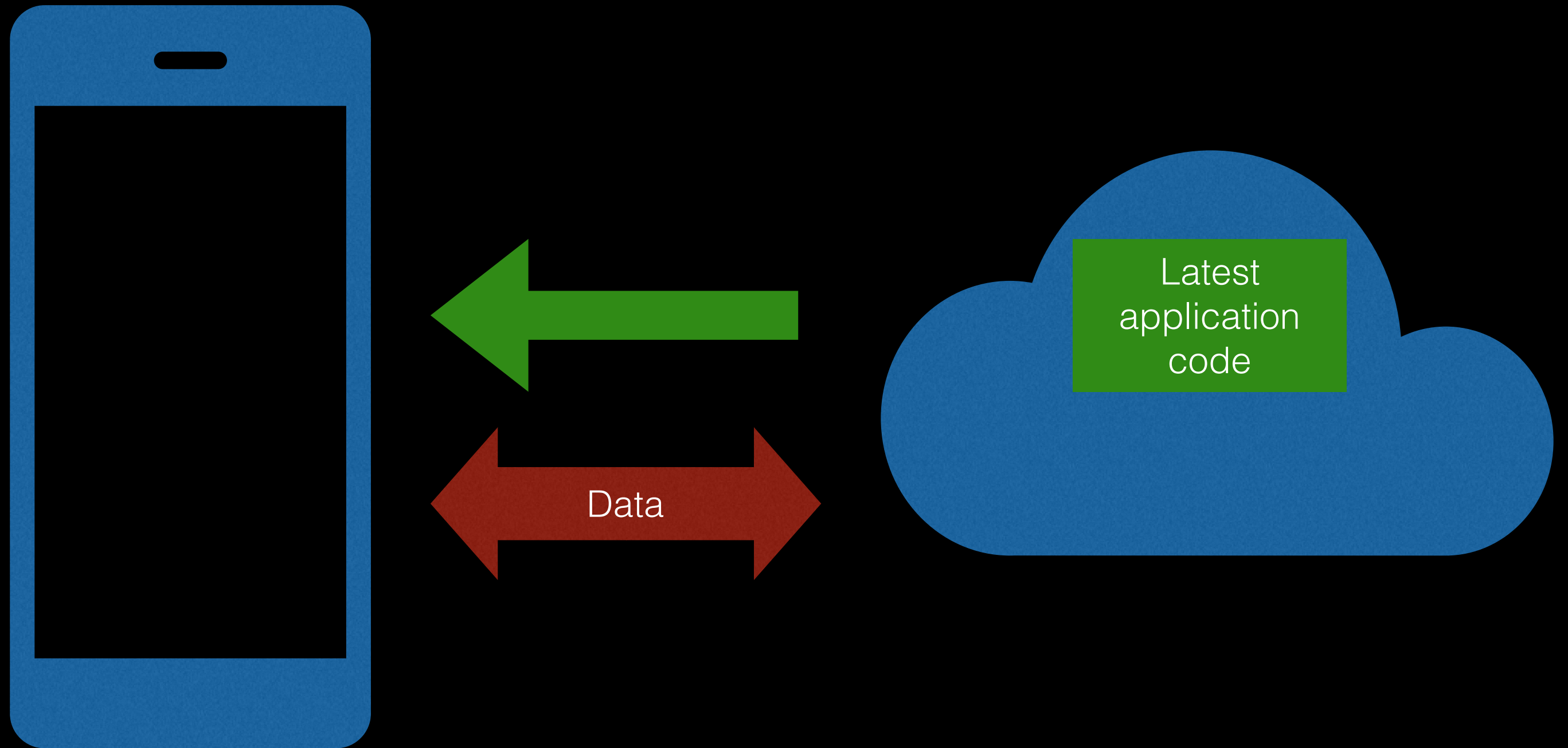
Comparing Approaches



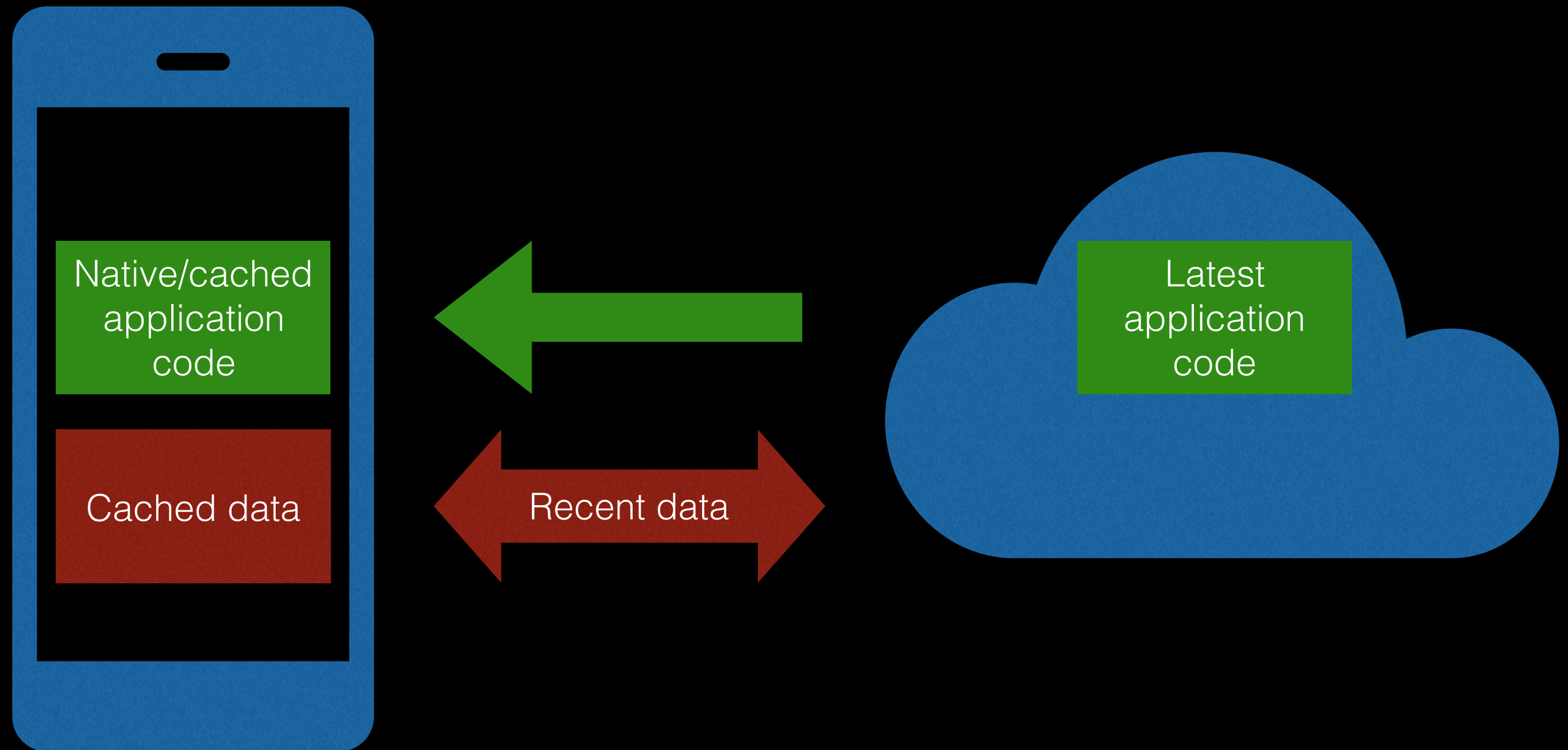
Native Apps



Web Apps



Hybrid Apps



Tradeoffs

Build

Test

Deploy

Launch

Testing to ensure things are
working as expected

Why write tests?

To know when something is broken

To get good signal from test failures to help find and fix bugs

To document the intent of the code

To help drive readable, maintainable code design

A/B Testing

A/B Testing Case Study

Generate a hypothesis

Implement an experiment

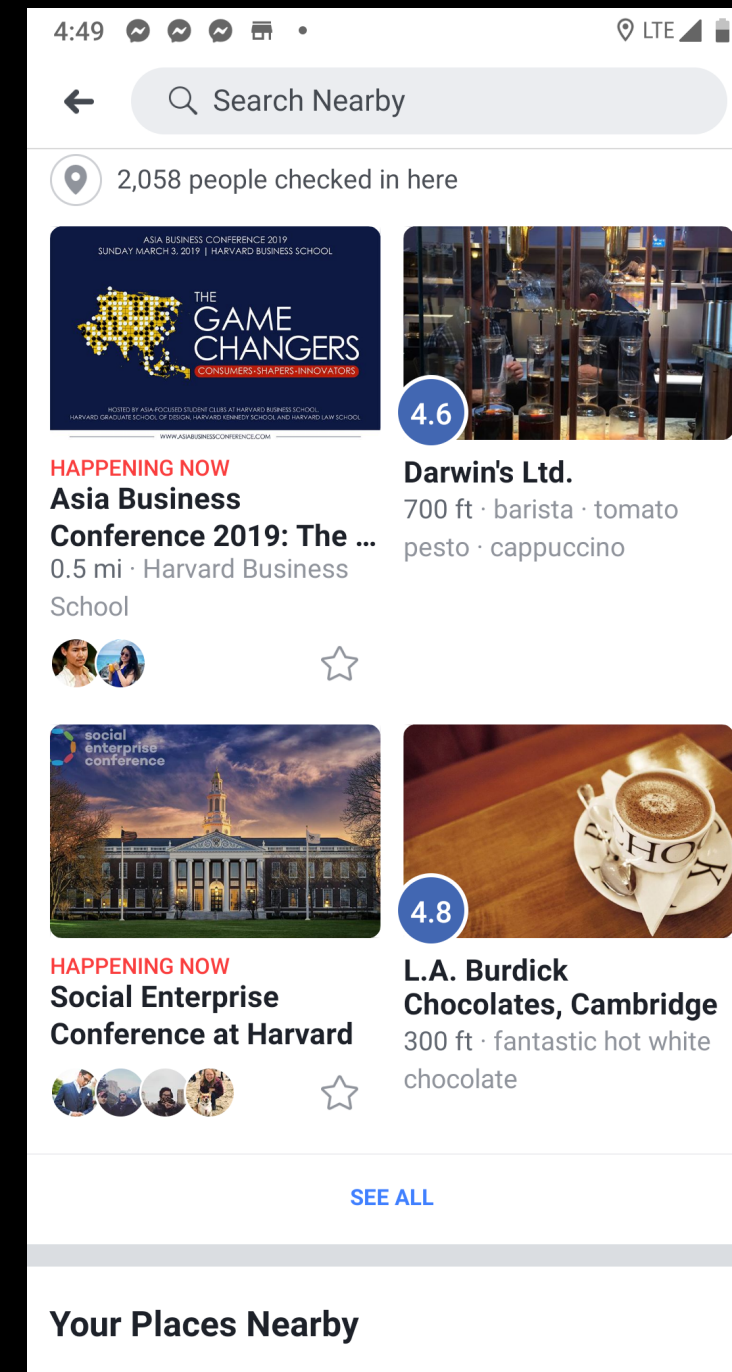
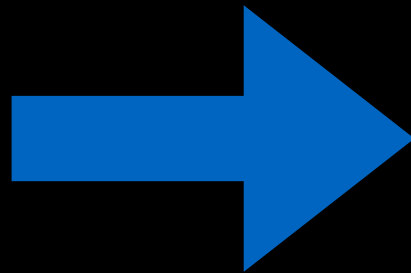
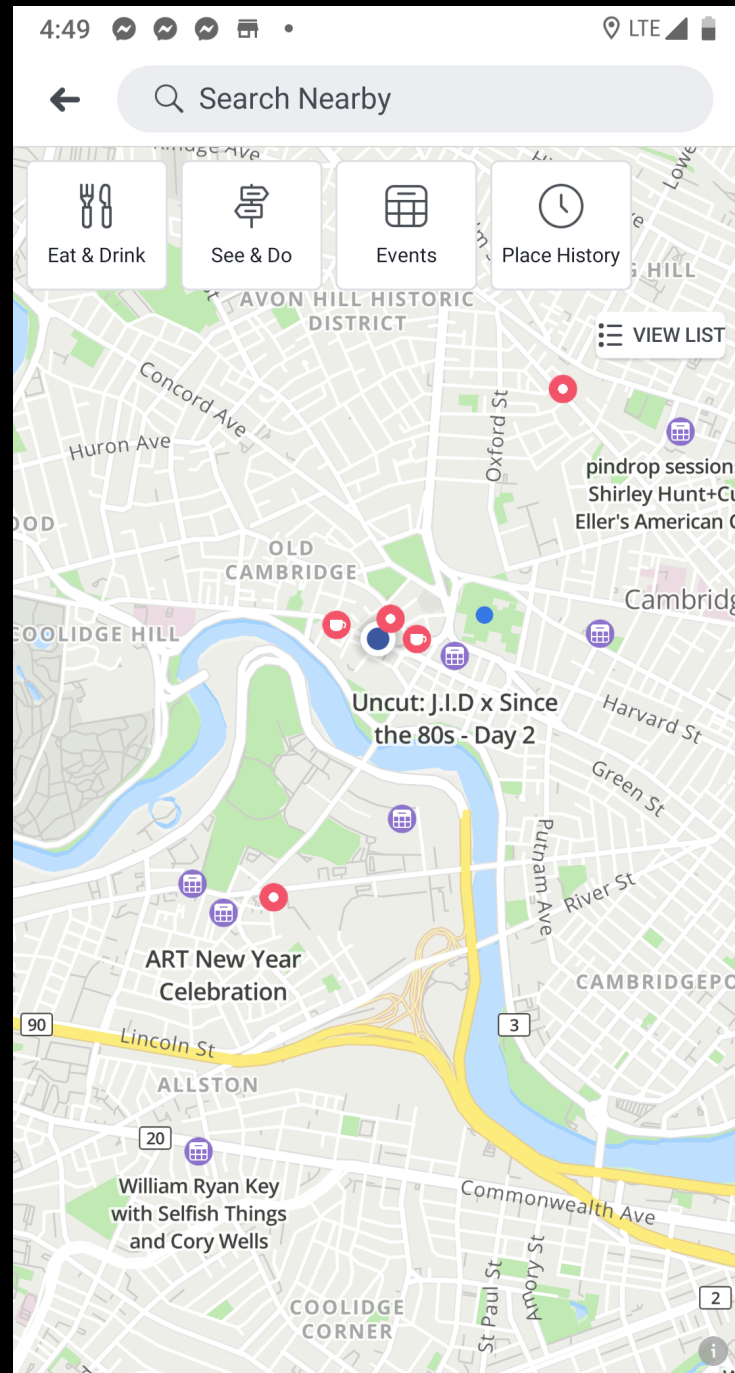
Review impact

Generate a hypothesis

Implement an experiment

Review impact

Local Search



4:49 [notifications] [calendar] [LTE] [signal] [battery]

← Search Nearby

📍 2,058 people checked in here

ASIA BUSINESS CONFERENCE 2019
SUNDAY MARCH 5, 2019 | HARVARD BUSINESS SCHOOL

THE GAME CHANGERS
CONSUMERS · SHAPERS · INNOVATORS

HOSTED BY ASIA FOCUSED STUDENT CLUB AT HARVARD BUSINESS SCHOOL,
HARVARD GRADUATE SCHOOL OF EDUCATION, HARVARD MEDICAL SCHOOL, AND HARVARD LAW SCHOOL.

WWW.ASIBUSINESSCONFERENCE.COM

4.6

HAPPENING NOW
Asia Business Conference 2019: The ...
0.5 mi · Harvard Business School

☆

Darwin's Ltd.
700 ft · barista · tomato pesto · cappuccino

social enterprise conference

4.8

HAPPENING NOW
Social Enterprise Conference at Harvard

☆

L.A. Burdick Chocolates, Cambridge
300 ft · fantastic hot white chocolate

[SEE ALL](#)

Your Places Nearby

A

4:50 [notifications] [calendar] [LTE] [signal] [battery]

← Social Enterprise Conference a... 🔍

Friends Who Are Going

- Alex Hem

Friends Who Are Interested

- Abdur Rehman
- Benny Pleat
- Megan White

B

“There’s too much friction to see
who’s attending”

“People care more about who’s
going than the event details”

“People expect UI elements to
be tappable”

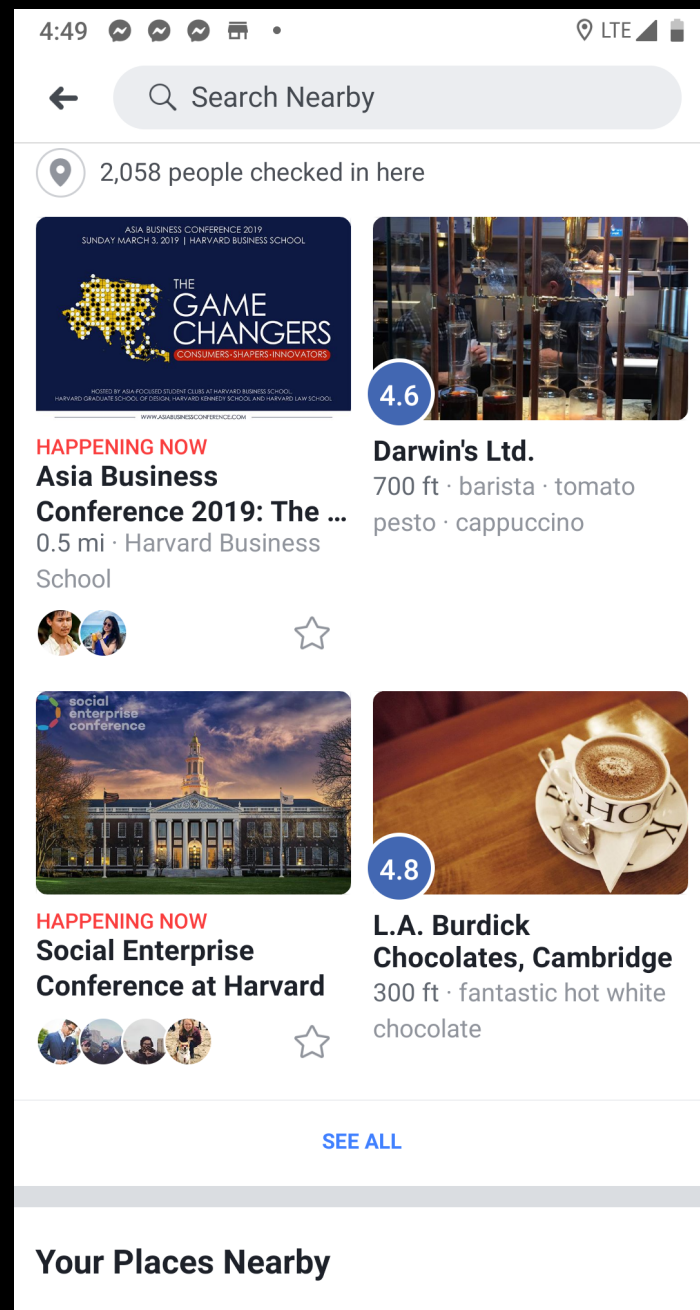
“Floating heads are trendy
right now”

Generate a hypothesis

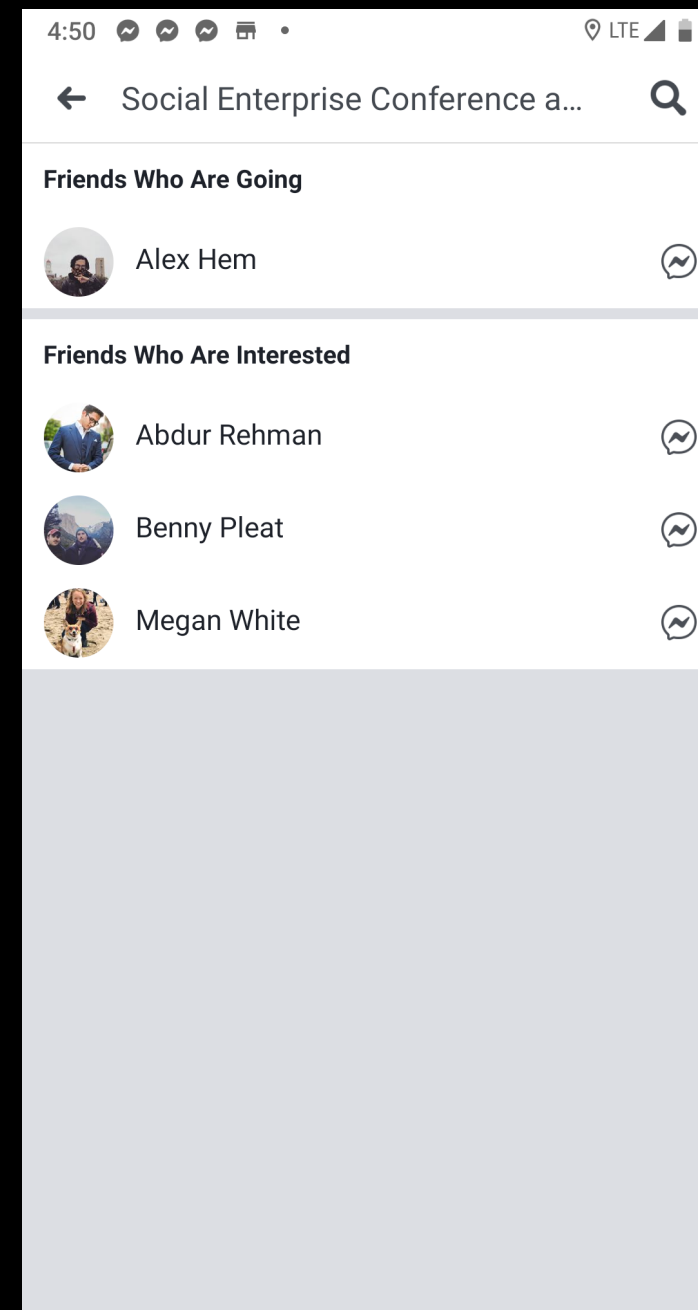
Implement an experiment

Review impact

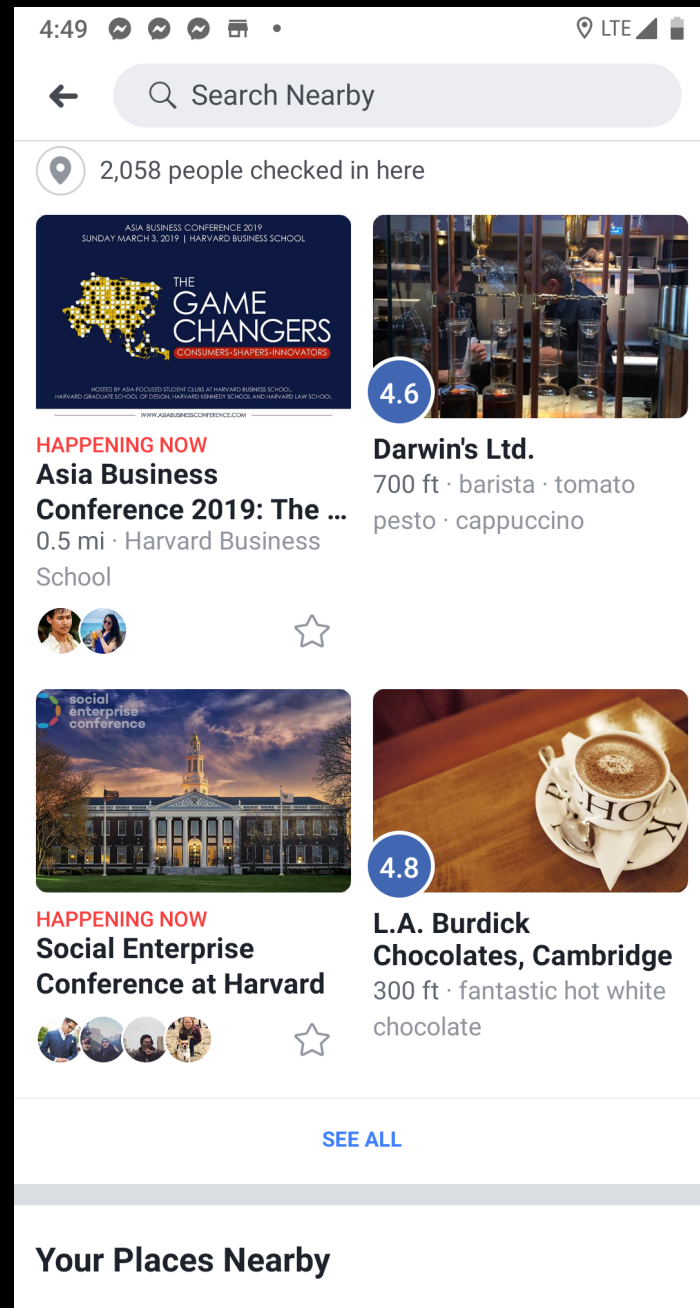
	Test	Control
Experiment size	10 million users in the US	10 million users in the US
Test timeframe	1 week	1 week



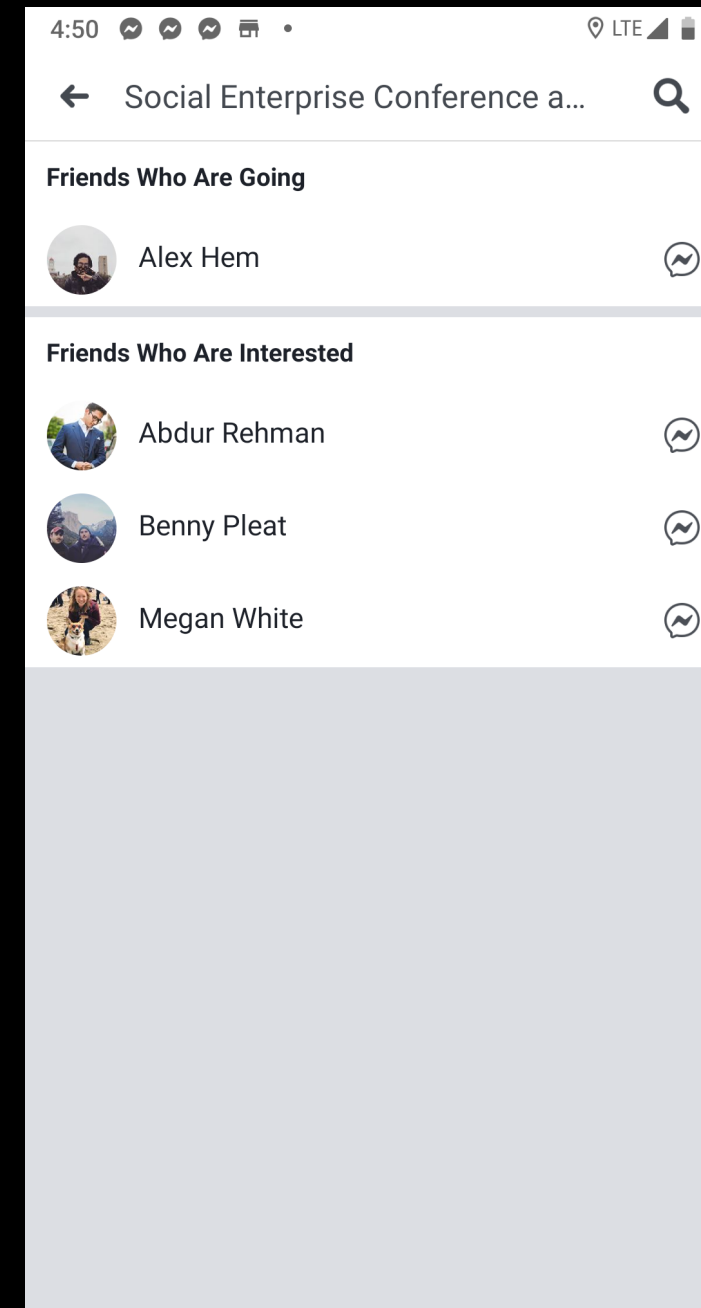
A



B



50%



50%

```
define_experiment(  
    'tappable_heads',  
    buckets=[  
        (range(0, 50), True),  
        (range(50, 100), False)  
    ]  
)
```



```
def render():  
    if experiment.get('tappable_heads'):  
        return render_tappable_heads()  
    else:  
        return render_heads()
```

Generate a hypothesis

Implement an experiment

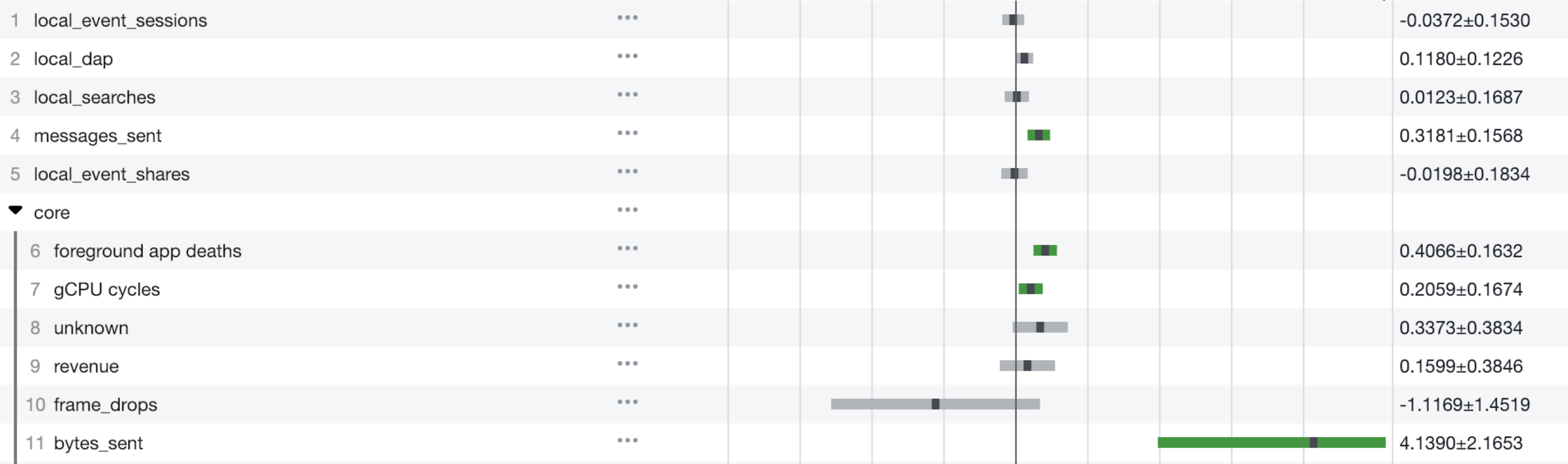
Review impact

Metrics

- Engagement?
- Performance?
- Cost?
- Complexity?

-4% -3% -2% -1% 0% 1% 2% 3% 4%

🔍 Delta % ⋮



Tradeoffs

Build

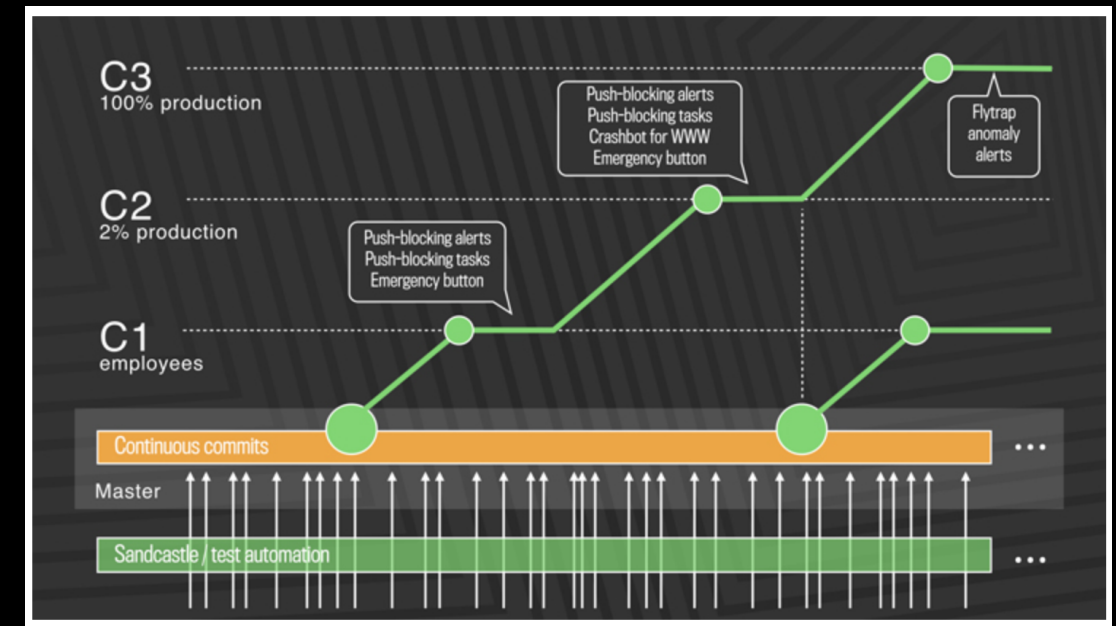
Test

Deploy

Launch

Deployment Strategies

SHIP IT!



Manual

Scheduled

Continuous

Deployment Strategies

[Small, Essential B2B Company]

- Provides cloud-based records management as a service to police and fire departments in the US
- SLA as part of their contracts that the site will have an uptime with no major bugs of 99%
- No more than 3000 users on their service every day, with high expectations of reliability
- 50 engineers

[Large, Consumer Social Company]

- Users get a free service on which they see ads. Users are more tolerant of downtime and have multiple alternatives when downtime on this service occurs. They're "sticky" users.
- ~6000 engineers
- The consumer product competes across all geographies and languages in a space with low barrier to entry.

Deployment at [Important B2B Company]

Every Tuesday:

1. The latest code from the “master” branch of the codebase will get “cut.”
2. For 4 work days, the code will then get released to an internal site using “mocked” data so that **QA** testers and engineers can test the site. Bugs are filed and fixed at this point.
3. For 1 work day, the code will then get released to a staging environment using real data, more tests follow.
4. One week from the original “cut” the newest features get released to 100% of users. Forward-deployed engineers follow-up with integration and customer support of the new features.

Deployment at [Consumer Social Company]

Roughly every 3 hours:

1. The “master” branch will get “cut”.
2. The code will run through a series of fully-automated tests (ie no QA engineers), if there are features or code changes that fail tests, they will either cause the release to stop or get removed from the release.
3. The code will then go out to
 - a. C1 (employees) for 30 minutes,
 - b. C2 (2% of users) for 1.5 hours,
 - c. and then C3 (98% of users), which takes about 45 minutes.
4. Any bugs that aren't caught in C1 or C2 and are in production are fixed by the next deployment, ie there are no concepts of “patches.”

Build

Test

Deploy

Launch

Launching

Craft

Simplicity

Ruthless Prioritization

“If we give ourselves more time, we’ll be tempted to add more scope” - Mike Krieger

After Launch

Launching is the easy part

How do you transition from the intensity of shipping to the sustained focus needed to succeed in the long run?

Balance Challenges

Adding New Features vs Stability & Performance

Burndown list of feature work

@mentions stickers



Adding New Features vs Stability & Performance

Include quality in roadmap - dedicated performance & quality sprints

Performance projects

- Media fetching based on user behavior

Structured Priorities vs Opportunities For Creativity

Structure that helps plan and distribute work, but still allows flexibility to pursue innovative new ideas

Opportunity to influence the roadmap continually infuses the product with fresh ideas

Sticker pinning

- Rewind
- Boomerang

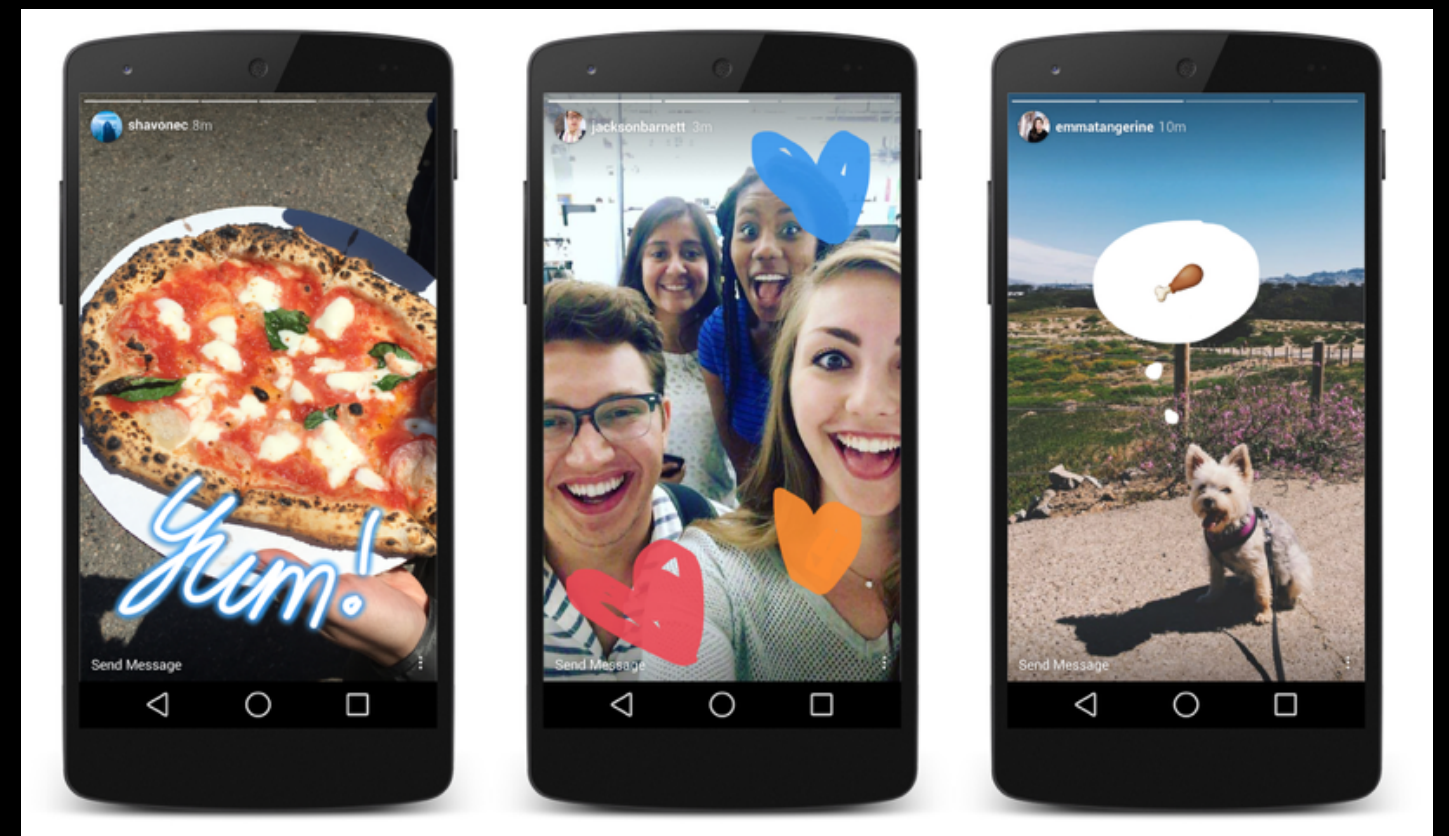


Moving Fast vs Craftsmanship

“Do the simple thing first”

How does that translate to V2, V3, and V(n)?

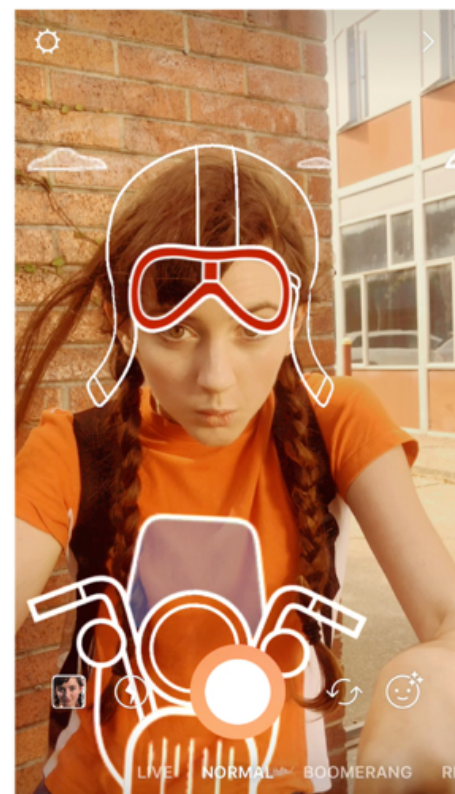
By adding complexity in layers, maintain the velocity to deliver a steady stream of improvements to the product



Maintaining Core Values vs Expanding the Team

Grow at a sustainable pace to ensure values are maintained and enhanced as new people join

Collaboration with other teams for face effects in Stories



Q&A

CS50 for MBAs

Software Engineering