



SOFTWARE DEVELOPMENT CONFERENCE

YOW! LONDON 2022

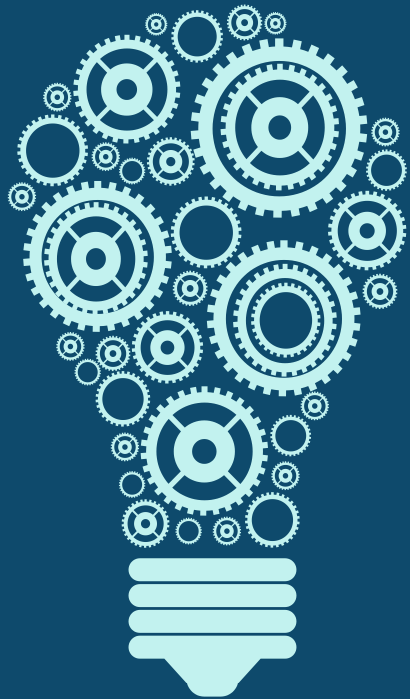
GOTO Guide App

- Download the app
- Ask questions
- Rate sessions

Doubling Engineering Productivity at eBay

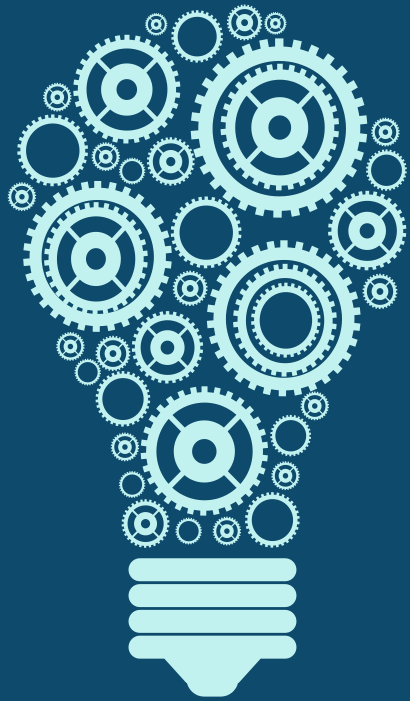
YOW London
November 2022
Randy Shoup

Problem Statement



*eBay's overall product velocity lags
industry leaders*

Problem Statement → Mission



eBay's overall product velocity lags industry leaders



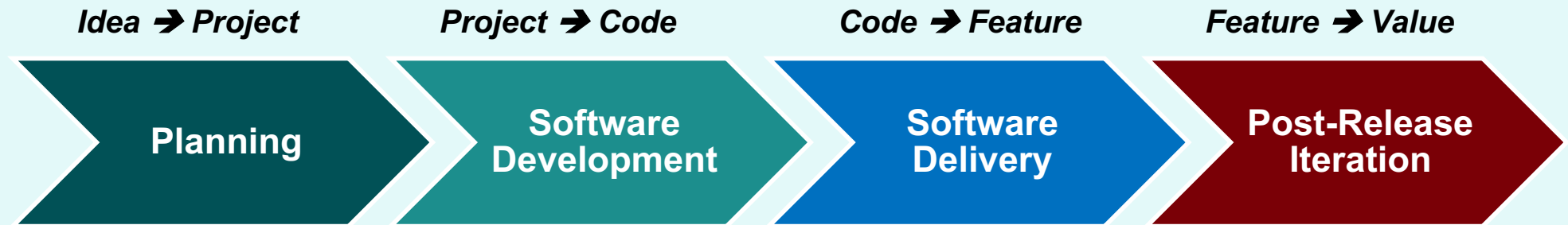
Make eBay's product velocity a competitive advantage

Why Are We Here?

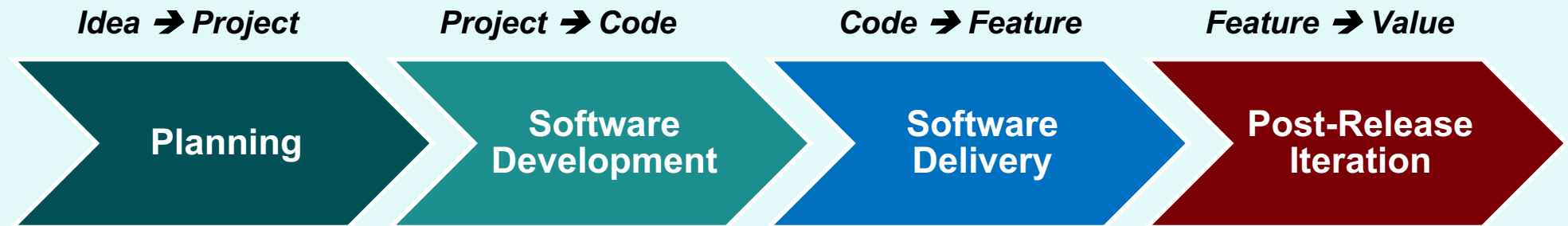


Assessment

Product Life Cycle

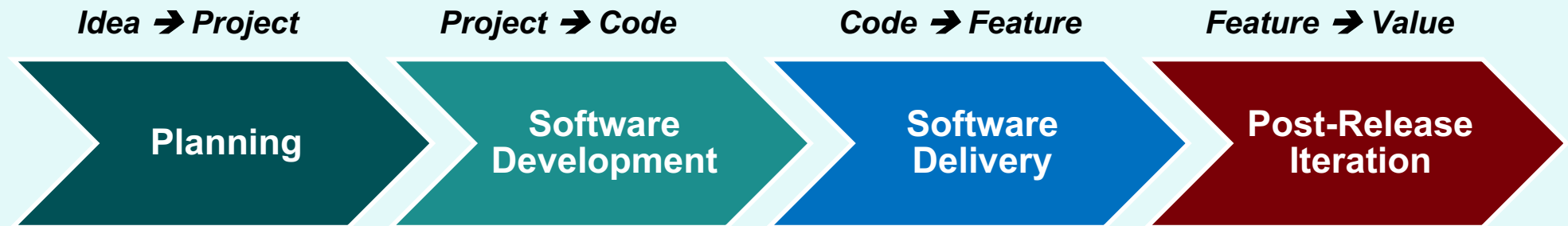


Product Life Cycle



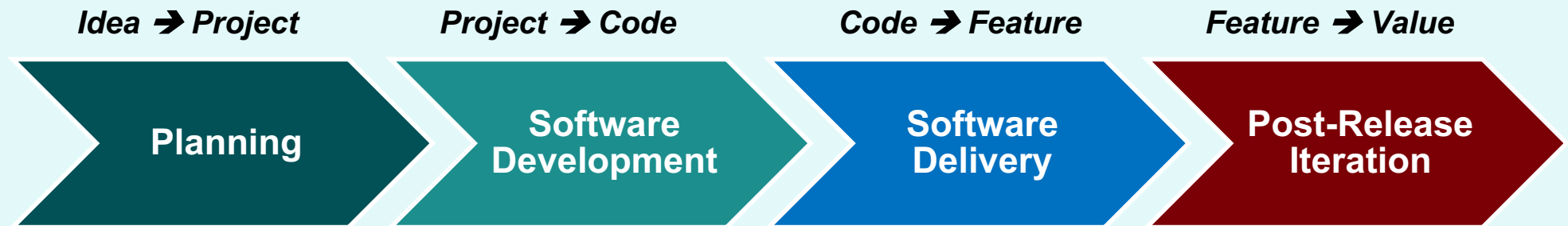
- Coordination
- Dependencies
- Too Much WIP

Product Life Cycle



- Coordination
 - Dependencies
 - Too Much WIP
- Build & Test Time
 - Context Switching & Wait States
 - Coupled Architecture
 - No Service Contracts
 - Hidden Work

Product Life Cycle

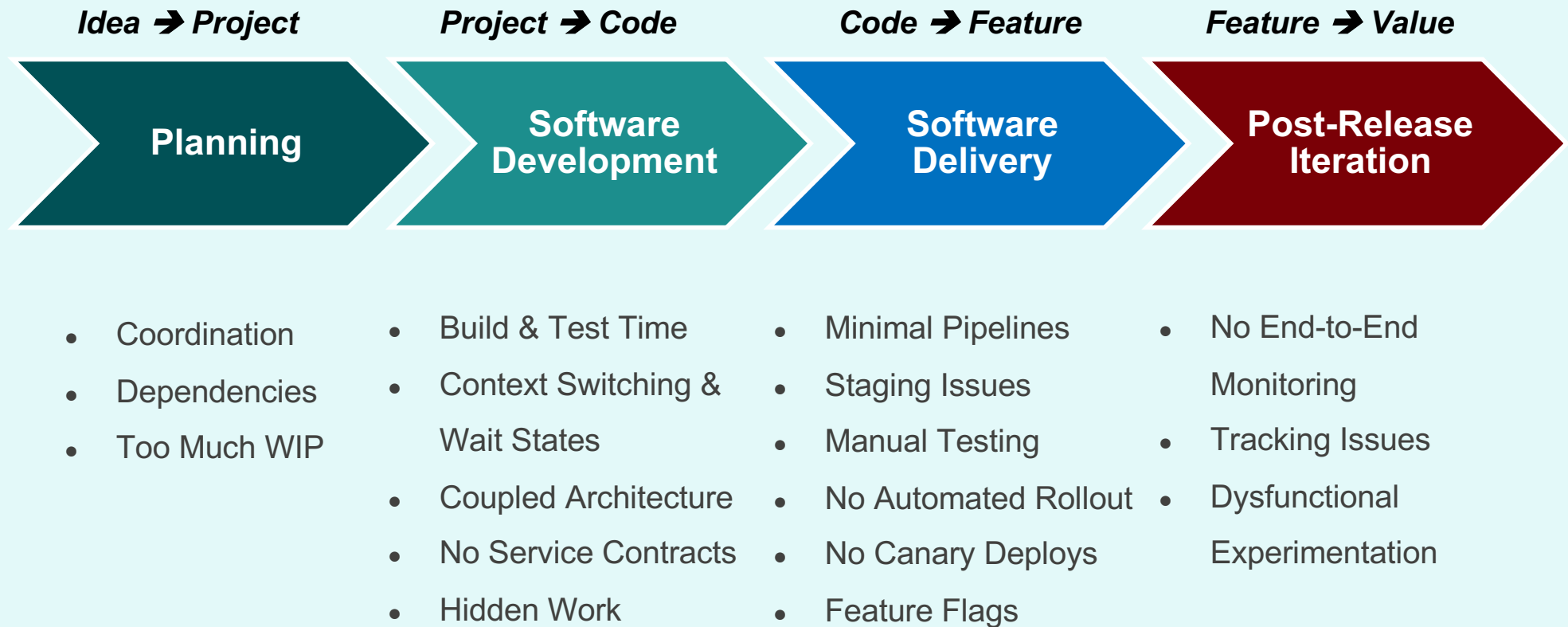


- Coordination
- Dependencies
- Too Much WIP

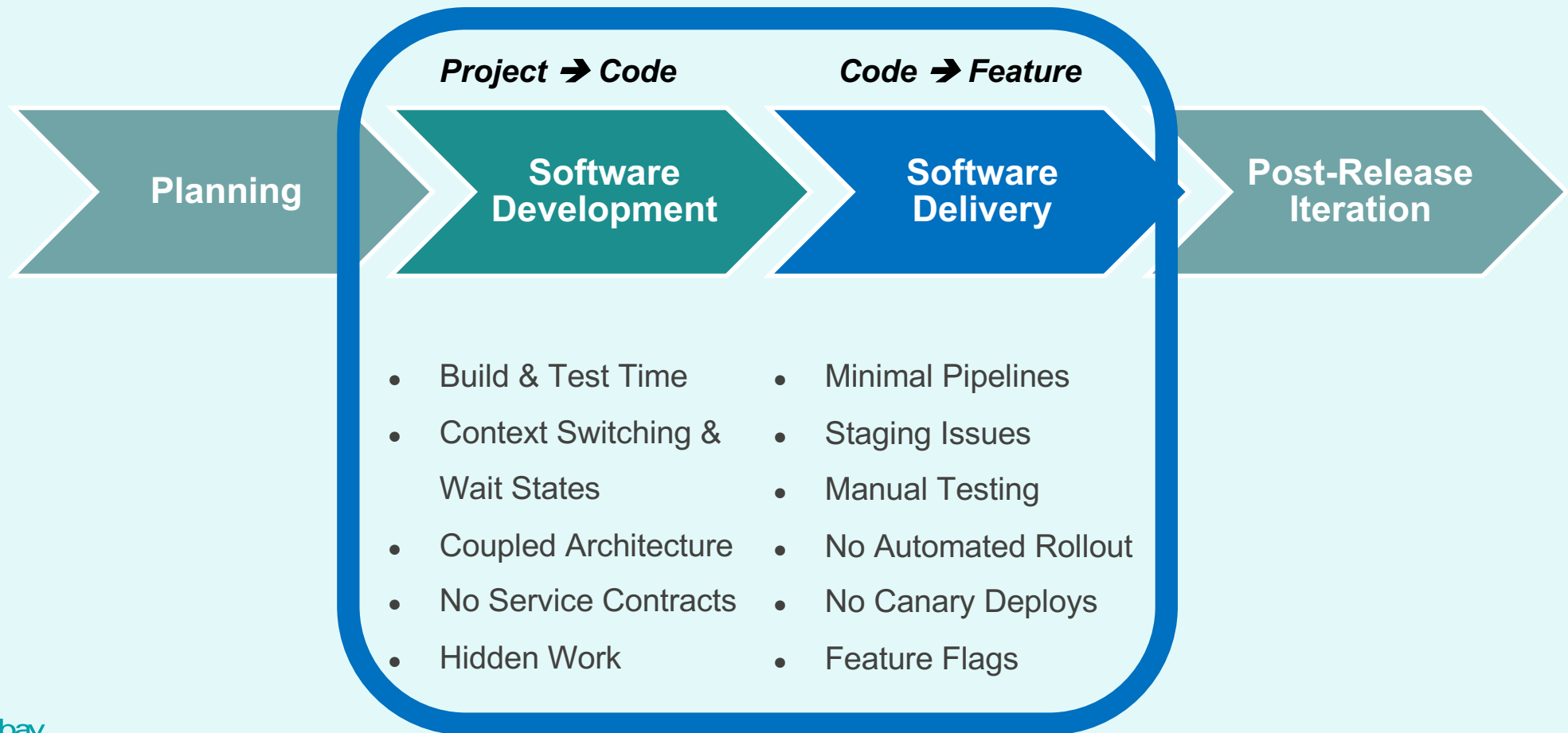
- Build & Test Time
- Context Switching & Wait States
- Coupled Architecture
- No Service Contracts
- Hidden Work

- Minimal Pipelines
- Staging Issues
- Manual Testing
- No Automated Rollout
- No Canary Deploys
- Feature Flags

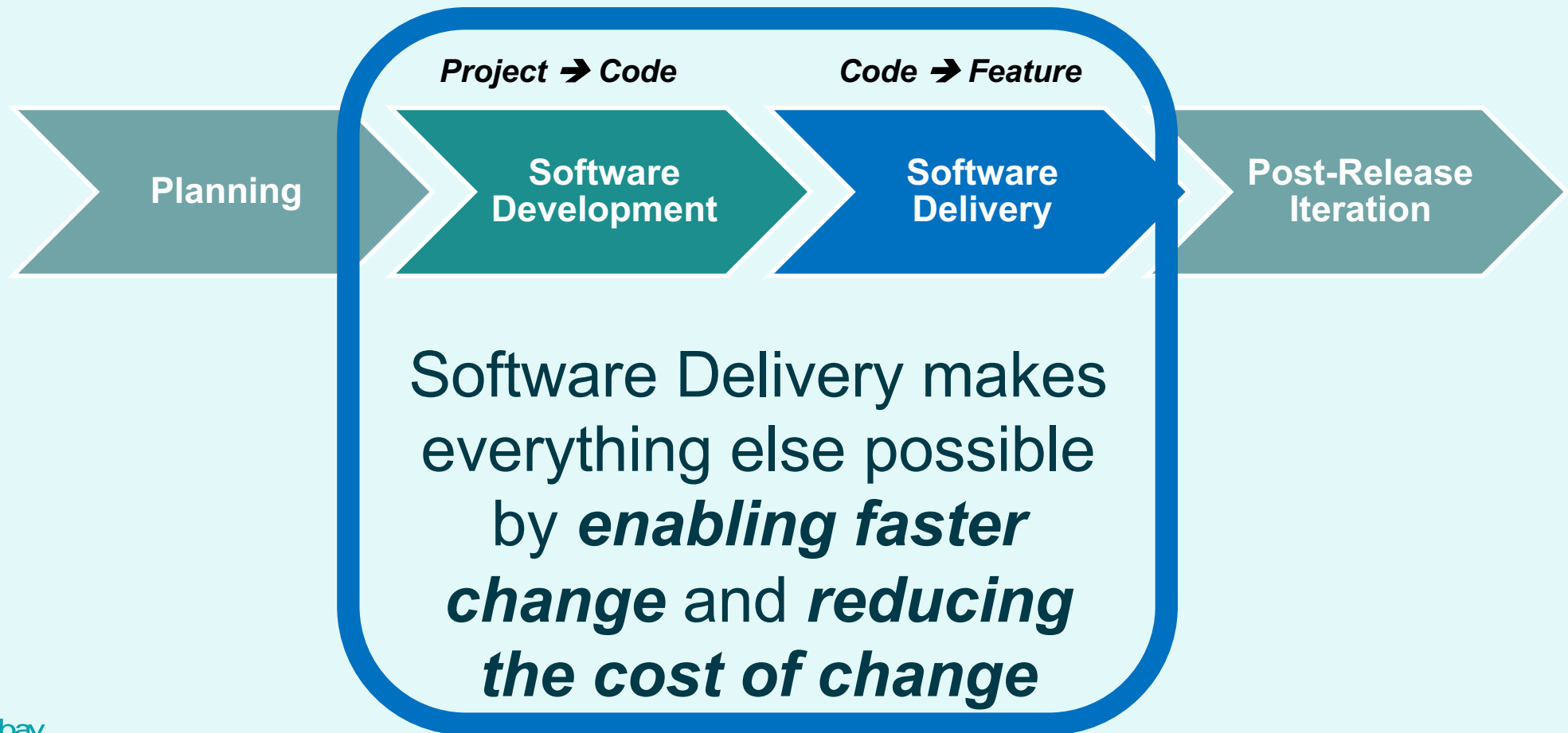
Product Life Cycle



Product Life Cycle

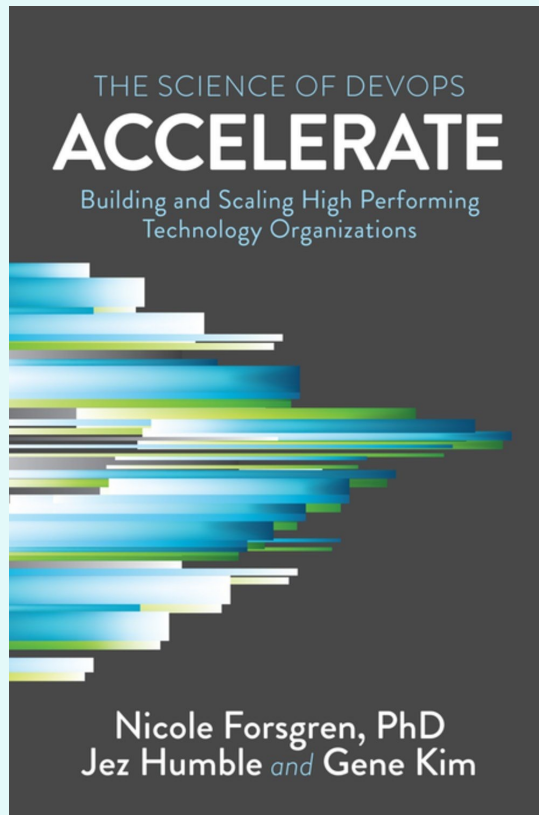


Product Life Cycle



Measuring Software Delivery

Software Delivery: State of DevOps Research

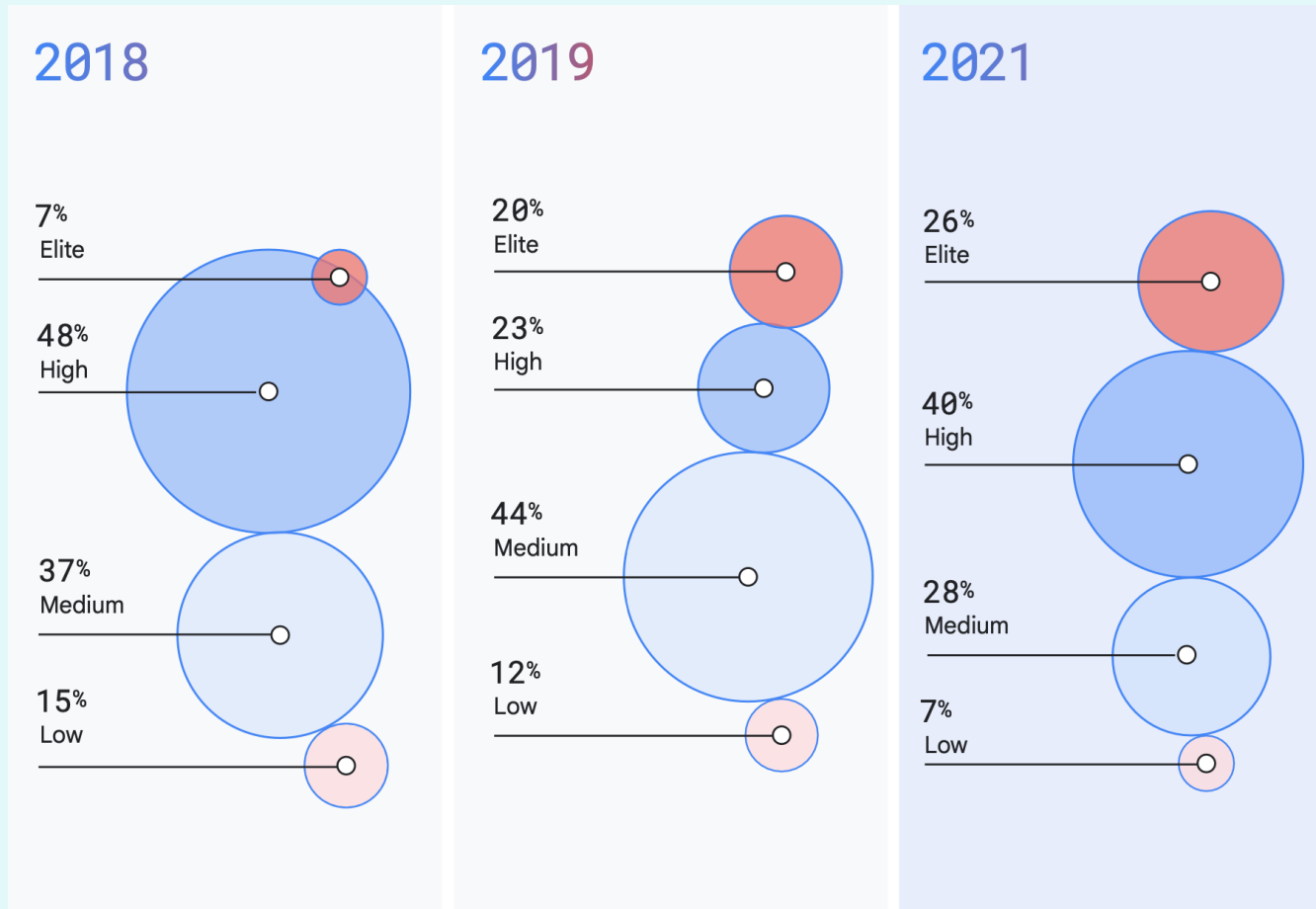


State of DevOps Surveys

- 9 yearly surveys from 2014-2022
- 31,000 survey responses
- Rigorous scientific methodology

Summarized in *Accelerate*

Software Delivery: Performance Clusters



Accelerate Metrics

Aspect of Software Delivery Performance*	Elite	High	Medium	Low
Deployment frequency For the primary application or service you work on, how often does your organization deploy code to production or release it to end users?	On-demand (multiple deploys per day)	Between once per day and once per week	Between once per week and once per month	Between once per month and once every six months
Lead time for changes For the primary application or service you work on, what is your lead time for changes (i.e., how long does it take to go from code committed to code successfully running in production)?	Less than one day	Between one day and one week	Between one week and one month	Between one month and six months
Time to restore service For the primary application or service you work on, how long does it generally take to restore service when a service incident or a defect that impacts users occurs (e.g., unplanned outage or service impairment)?	Less than one hour	Less than one day ^a	Less than one day ^a	Between one week and one month
Change failure rate For the primary application or service you work on, what percentage of changes to production or released to users result in degraded service (e.g., lead to service impairment or service outage) and subsequently require remediation (e.g., require a hotfix, rollback, fix forward, patch)?	0-15% ^{b,c}	0-15% ^{b,d}	0-15% ^{c,d}	46-60%

Accelerate Metrics

Aspect of Software Delivery Performance*	Elite	High	Medium	Low
Deployment frequency For the primary application or service you work on, how often does your organization deploy code to production or release it to end users?	On-demand (multiple deploys per day)	Between once per day and once per week	Between once per week and once per month	Between once per month and once every six months
Lead time for changes For the primary application or service you work on, what is your lead time for changes (i.e., how long does it take to go from code committed to code successfully running in production)?	Less than one day	Between one day and one week	Between one week and one month	Between one month and six months
Time to restore service For the primary application or service you work on, how long does it generally take to restore service when a service incident or a defect that impacts users occurs (e.g., unplanned outage or service impairment)?	Less than one hour	Less than one day ^a	Less than one day ^a	Between one week and one month
Change failure rate For the primary application or service you work on, what percentage of changes to production or released to users result in degraded service (e.g., lead to service impairment or service outage) and subsequently require remediation (e.g., require a hotfix, rollback, fix forward, patch)?	0-15% ^{b,c}	0-15% ^{b,d}	0-15% ^{c,d}	46-60%

Months

Accelerate Metrics

Aspect of Software Delivery Performance*	Elite	High	Medium	Low
Deployment frequency For the primary application or service you work on, how often does your organization deploy code to production or release it to end users?	On-demand (multiple deploys per day)	Between once per day and once per week	Between once per week and once per month	Between once per month and once every six months
Lead time for changes For the primary application or service you work on, what is your lead time for changes (i.e., how long does it take to go from code committed to code successfully running in production)?	Less than one day	Between one day and one week	Between one week and one month	Between one month and six months
Time to restore service For the primary application or service you work on, how long does it generally take to restore service when a service incident or a defect that impacts users occurs (e.g., unplanned outage or service impairment)?	Less than one hour	Less than one day ^a	Less than one day	Between one week and one month
Change failure rate For the primary application or service you work on, what percentage of changes to production or released to users result in degraded service (e.g., lead to service impairment or service outage) and subsequently require remediation (e.g., require a hotfix, rollback, fix forward, patch)?	0-15% ^{b,c}	0-15% ^{b,d}	0-15% ^{c,d}	16-60%

Weeks

Accelerate Metrics

Aspect of Software Delivery Performance*	Elite	High	Medium	Low
Deployment frequency For the primary application or service you work on, how often does your organization deploy code to production or release it to end users?	On-demand (multiple deploys per day)	Between once per day and once per week	Between once per week and once per month	Between once per month and once every six months
Lead time for changes For the primary application or service you work on, what is your lead time for changes (i.e., how long does it take to go from code committed to code successfully running in production)?	Less than one day	Between one day and one week	Between one week and one month	Between one month and six months
Time to restore service For the primary application or service you work on, how long does it generally take to restore service when a service incident or a defect that impacts users occurs (e.g., unplanned outage or service impairment)?	Less than one hour	Less than one day ^a	Less than one day ^a	Between one week and one month
Change failure rate For the primary application or service you work on, what percentage of changes to production or released to users result in degraded service (e.g., lead to service impairment or service outage) and subsequently require remediation (e.g., require a hotfix, rollback, fix forward, patch)?	0-15% ^{b,c}	0-15% ^{b,d}	0-15% ^{c,d}	46-60%

Accelerate Metrics

Aspect of Software Delivery Performance*	Elite	High	Medium	Low
Deployment frequency For the primary application or service you work on, how often does your organization deploy code to production or release it to end users?	On-demand (multiple deploys per day)	Between once per day and once per week	Between once per week and once per month	Between once per month and once every six months
Lead time for changes For the primary application or service you work on, what is your lead time for changes (i.e., how long does it take to go from code committed to code successfully running in production)?	Less than one day	Between one day and one week	Between one week and one month	Between one month and six months
Time to restore service For the primary application or service you work on, how long does it generally take to restore service when a service incident or a defect that impacts users occurs (e.g., unplanned outage or service impairment)?	Less than one hour	Less than one day ^a	Less than one day ^a	Between one week and one month
Change failure rate For the primary application or service you work on, what percentage of changes to production or released to users result in degraded service (e.g., lead to service impairment or service outage) and subsequently require remediation (e.g., require a hotfix, rollback, fix forward, patch)?	0-15% ^{b,c}	0-5% ^{b,d}	0-15% ^{c,d}	46-60%

Measuring eBay

Accelerate Metrics: eBay Overall

Aspect of Software Delivery Performance*	Elite	High	Medium	Low
Deployment frequency For the primary application or service you work on, how often does your organization deploy code to production or release it to end users?	On-demand (multiple deploys per day)	Between once per day and once per week	Between once per week and once per month	Between once per month and once every six months
Lead time for changes For the primary application or service you work on, what is your lead time for changes (i.e., how long does it take to go from code committed to code successfully running in production)?	Less than one day	Between one day and one week	Between one week and one month	Between one month and six months
Time to restore service For the primary application or service you work on, how long does it generally take to restore service when a service incident or a defect that impacts users occurs (e.g., unplanned outage or service impairment)?	Less than one hour	Less than one day ^a	Less than one day ^a	Between one week and one month
Change failure rate For the primary application or service you work on, what percentage of changes to production or released to users result in degraded service (e.g., lead to service impairment or service outage) and subsequently require remediation (e.g., require a hotfix, rollback, fix forward, patch)?	0-15% ^{b,c}	0-15% ^{b,d}	0-15% ^{c,d}	46-60%

Accelerate Metrics: Velocity Results

Aspect of Software Delivery Performance*	Elite	High	Medium	Low
Deployment frequency For the primary application or service you work on, how often does your organization deploy code to production or release it to end users?	On-demand (multiple deploys per day)	Between once per day and once per week	Between once per week and once per month	Between once per month and once every six months
Lead time for changes For the primary application or service you work on, what is your lead time for changes (i.e., how long does it take to go from code committed to code successfully running in production)?	Less than one day	Between one day and one week	Between one week and one month	Between one month and six months
Time to restore service For the primary application or service you work on, how long does it generally take to restore service when a service incident or a defect that impacts users occurs (e.g., unplanned outage or service impairment)?	Less than one hour	Less than one day ^a	Less than one day ^a	Between one week and one month
Change failure rate For the primary application or service you work on, what percentage of changes to production or released to users result in degraded service (e.g., lead to service impairment or service outage) and subsequently require remediation (e.g., require a hotfix, rollback, fix forward, patch)?	0-15% ^{b,c}	0-15% ^{b,d}	0-15% ^{c,d}	46-60%

— **Velocity Results:**

Doubled engineering productivity for pilot teams

(Holding team size and composition constant, deliver 2x features and bug fixes)

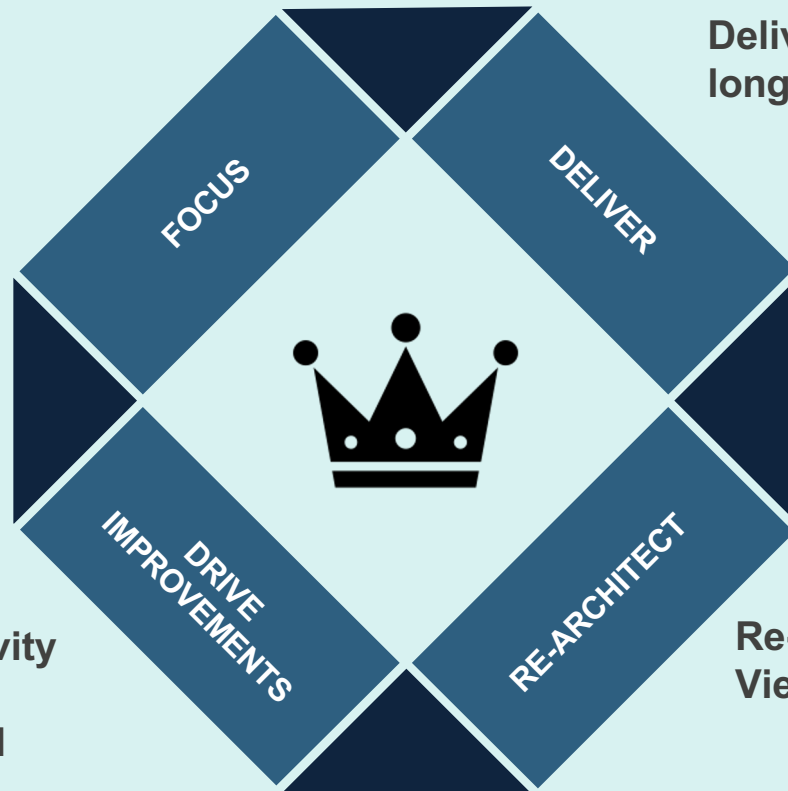
Velocity Initiative

Think Big, Start Small, Learn Fast

Focus on select

- Pilot Domains
- Pilot Applications
- Platform Tracks

Deliver short term wins and long term capabilities

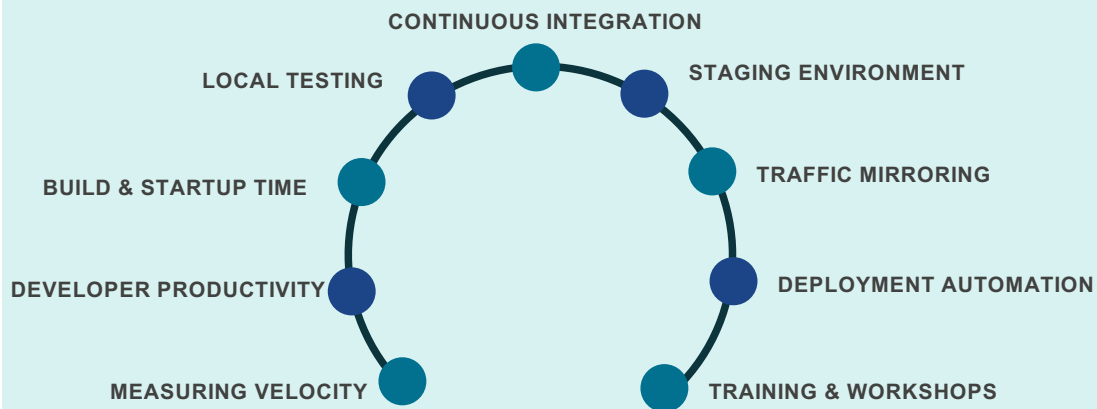


Drive improvements in

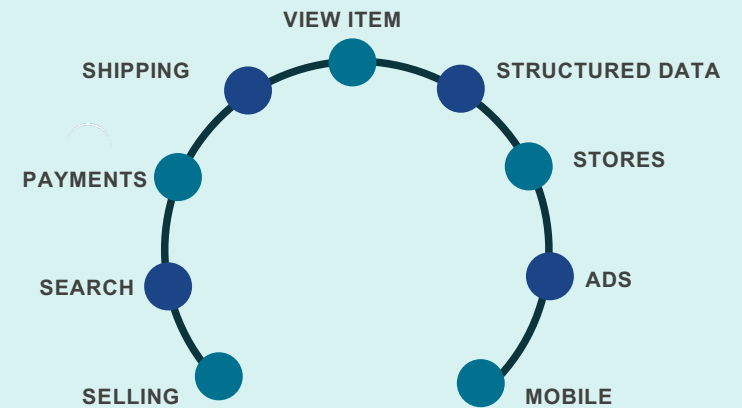
- Developer Productivity
- Software Delivery
- Instrumentation and Monitoring

Re-architect critical areas like View Item and Mobile

Platform Tracks



Pilot Domains



How We Work

COLLABORATE

- Cross-functional Leadership
- Embedding model
- Platform and Pilot teams working together

COMMUNICATE

- Daily leadership standups
- Weekly Team-of-Teams meeting
- Weekly Deep Dives with teams
- Monthly Operating Review

MEASURE

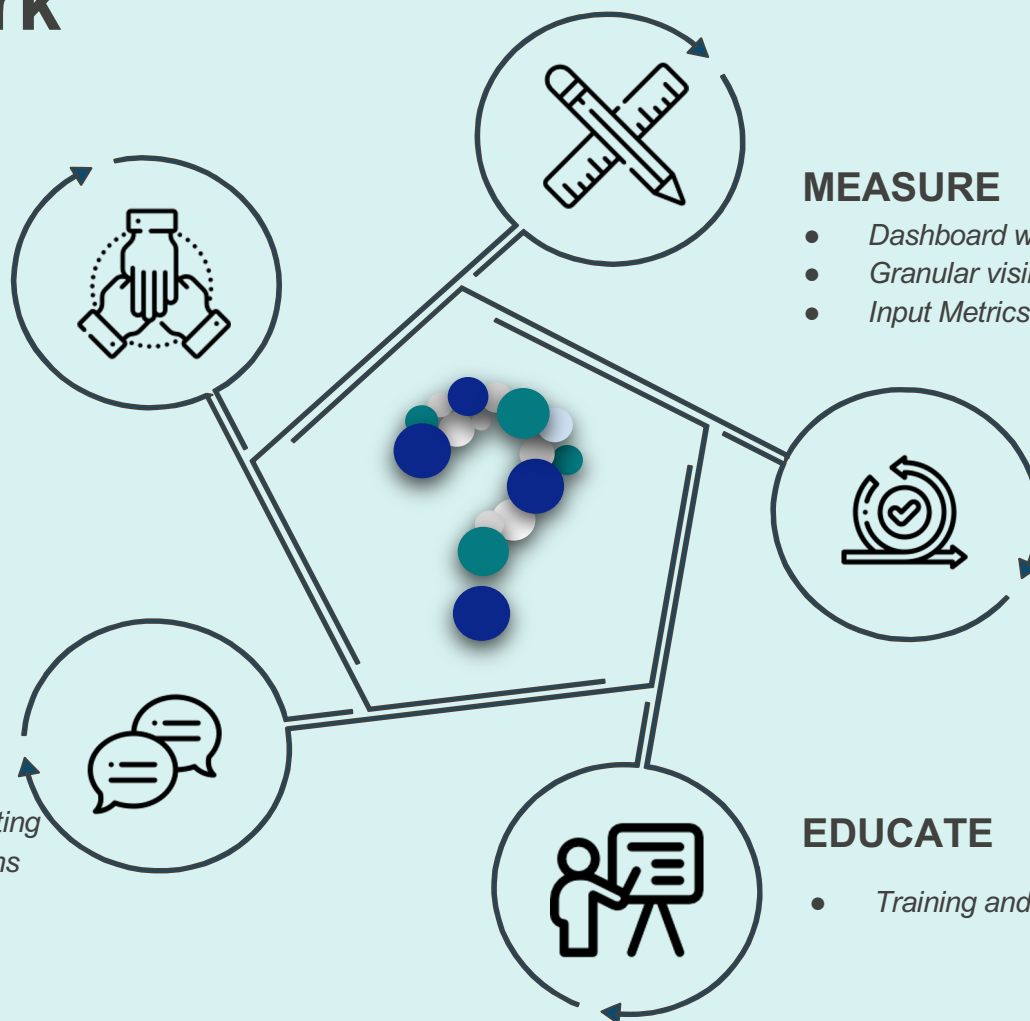
- Dashboard with Four Key Metrics for every app
- Granular visibility into entire delivery pipeline
- Input Metrics for each Platform Track

ITERATE

- Remove impediments to flow
- Tight Plan-Do-Check-Act cycles
- Stop effort if no improvement

EDUCATE

- Training and Workshops

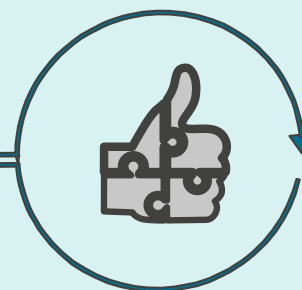
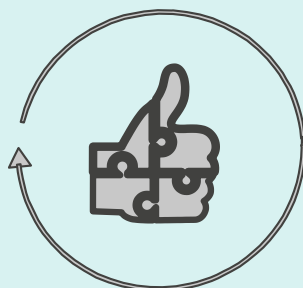


Initiative Results

DOUBLED PRODUCTIVITY OF VELOCITY PILOT TEAMS

Holding team size constant, teams deliver >2x the features

- *Pilot teams are 10% of actively-developed apps*
- *Deployment Frequency improved 5x*
- *Lead Time improved 5x*
- *Change Failure Rate improved 3x*
- *Time to Recover improved 3x*



WHAT WE DID

- *Focused on removing bottlenecks*
- *Reduced build, startup and PR validation times*
- *Invested heavily in Staging environment*
- *Automated upgrades, testing, deployment, Site Speed*
- *Streamlined team processes, code reviews, "Partner Signoffs"*
- *Moved from monthly to weekly mobile releases*

Culture & Behavior

EXCITEMENT & FUN

- Regular weekly progress on metrics
- Teams inspiring each other to improve
- Teams forced their way in to the program (!)
- Developers never want to go back to the old ways

COMMUNITY & SHARING

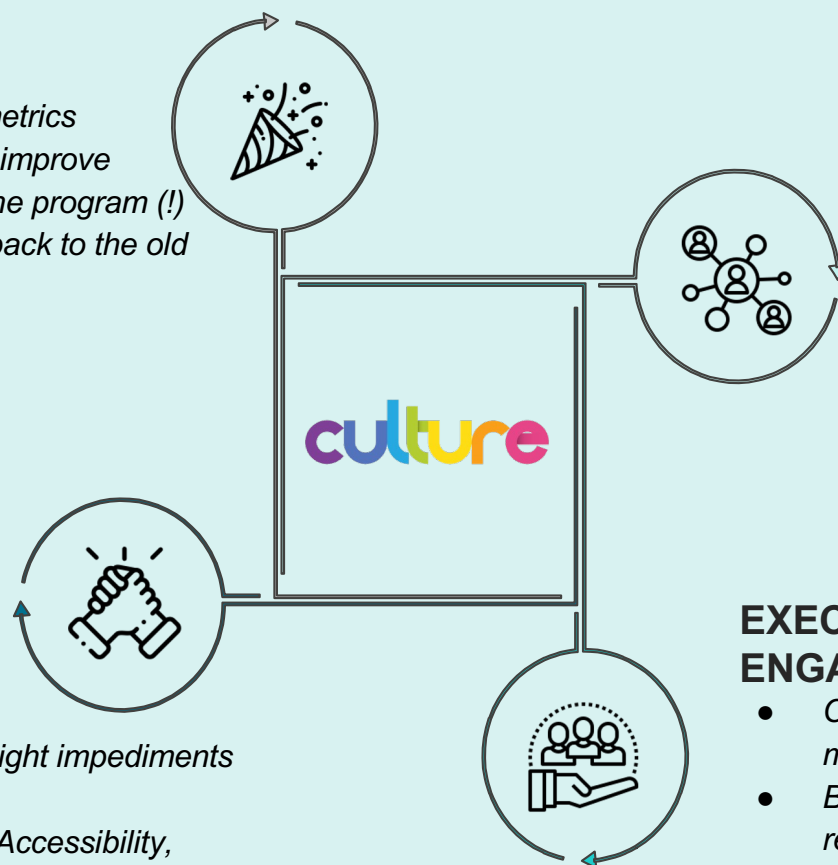
- Teams automating their own workflows (PR Reminders, Performance testing, Accessibility testing)
- Regular team demos of new tools and practices
- Sharing tools and learnings with each other

COLLABORATION

- Psychologically safe to highlight impediments outside team
- Partner with Security, SOX, Accessibility, Localization, etc.

EXECUTIVE SUPPORT & ENGAGEMENT

- CEO constantly highlights at All-Hands, exec meetings
- Board of Directors and Quarterly Earnings reports
- “The most important initiative at the company. Go faster!”



Scaling the Initiative

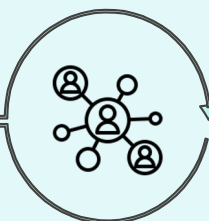
QUARTERLY COHORTS

- *Expand to 50%+ of active applications through 2022*
- *Oversubscribed already in Q1, Q2 cohorts*



SITE-WIDE UPGRADES

- *Patch security vulnerabilities, upgrade dependencies, retire legacy APIs, etc.*
- *Practice regular deployments for every app, even if not actively maintained*
- *Carrot and stick to build out automation*
- *Opinionated “patch pipeline” to make it simple and easy*



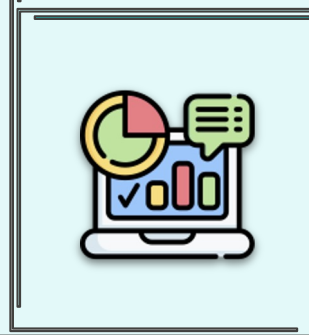
VELOCITY PLAYBOOK

- *Standardized “Paved Path” for teams to improve*
- *Problem – Solution model*
- *Collection of tools and best practices, with eBay examples*
- *Training videos, testimonials, team contacts*
- *Further resources*



MOBILE MODERNIZATION

- *Continue tightening weekly release schedule*
- *GraphQL*
- *SwiftUI, Jetpack Compose*



Ongoing Challenges

PROGRAM OUTCOMES

- *Improving team-level outputs, not overall eBay outcomes*

INITIATIVE TEAM

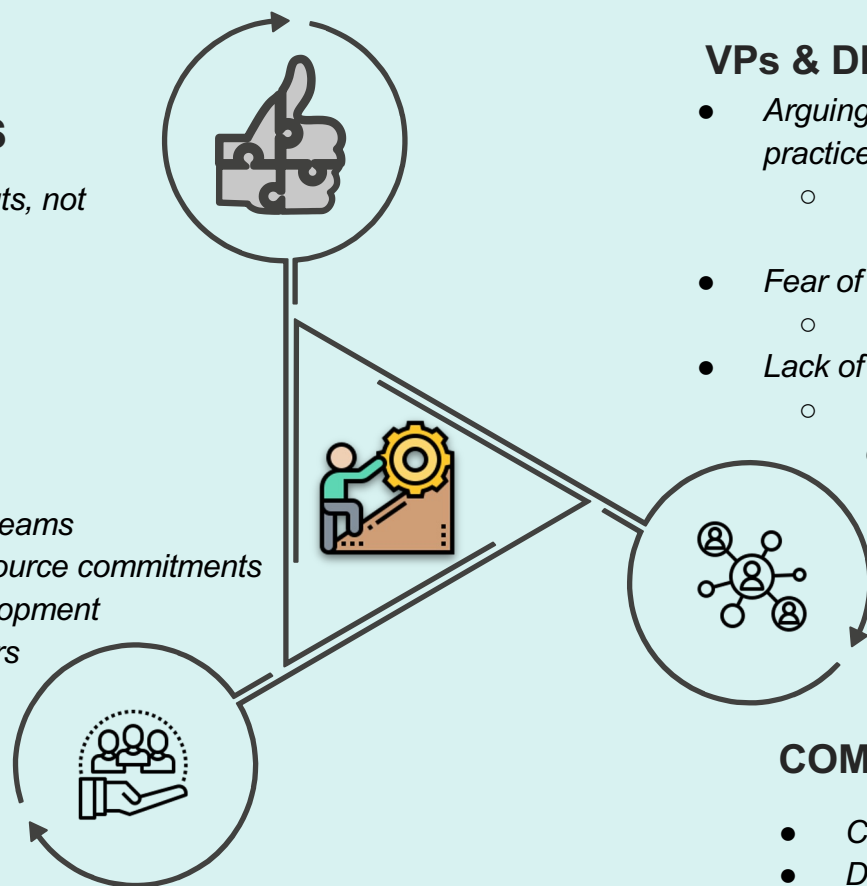
- *Under-resourced platform teams*
- *Ensuring product team resource commitments*
- *Often led by QE, not Development*
- *Overtaxed individual leaders*

VPs & DIRECTORS

- *Arguing about the metrics instead of improving the practices*
 - *“Adding new apps makes my average worse”*
- *Fear of failure and consequences*
 - *“Going too fast caused this bug”*
- *Lack of belief in program approach*
 - *“Validating quality slows us down; let’s stop doing that”*

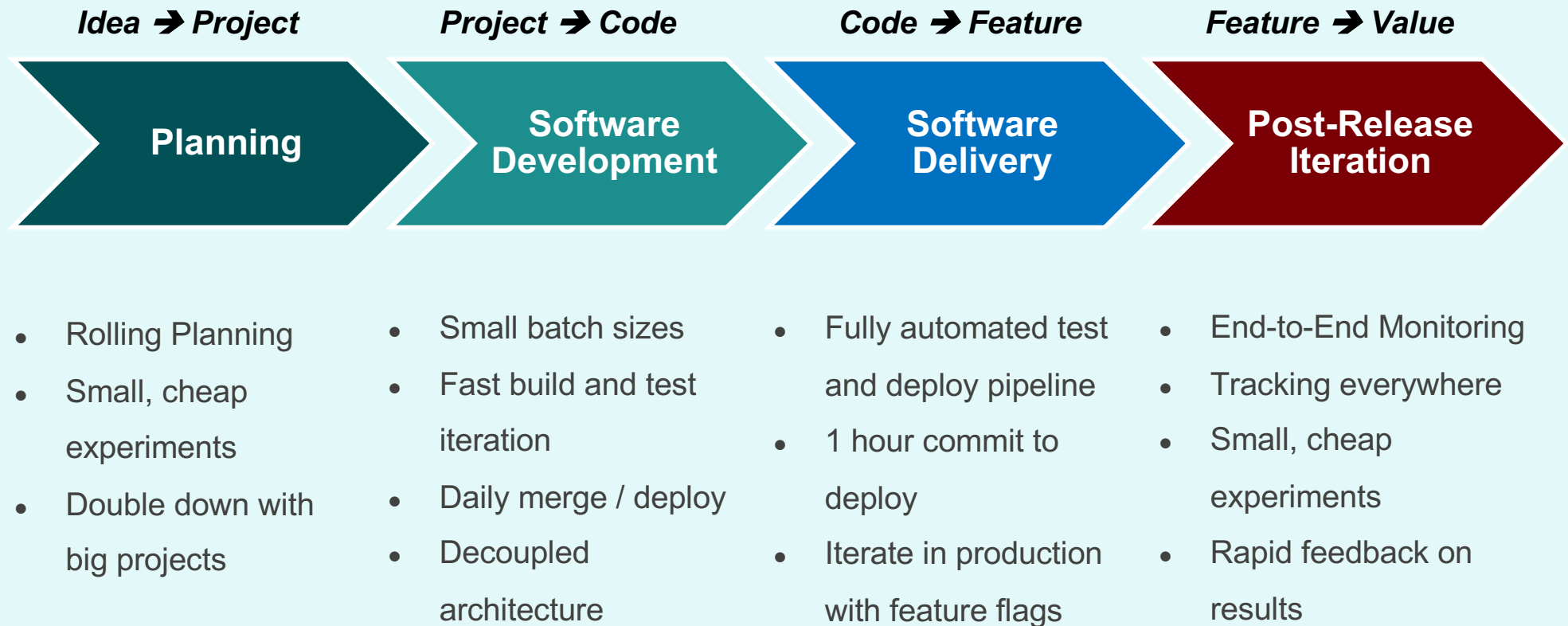
COMPANY “IMMUNE SYSTEM”

- *Changing mindset is hard*
- *Different incentives and behaviors*
- **81% of transformations fail (!)**



Future

Product Life Cycle: Future Goals





YOW!



Follow us **@goto_con**
Share your experience using **#YOWlondon**



Follow us **@GOTOcon**
Share your experience using **#YOWlondon**



Follow us **GOTO Conferences**
Join **322k+ subscribers & 36m+ views**

