

The Key to High Performance: What the Data Says

Nicole Forsgren, PhD

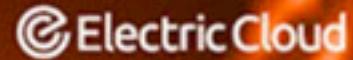
CEO and Chief Scientist, DevOps Research and Assessment (DORA)
Research Affiliate, Clemson University, Florida International University

The Data Behind DevOps: Becoming a High Performer

DevOps ENTERPRISE SUMMIT

#DOES14

PRESENTED BY:



IT REVOLUTION PRESS
HELPING SPARK THE IT CLOUD REVOLUTION

The journey



2014: DevOps works!



2015: IT goes lean.



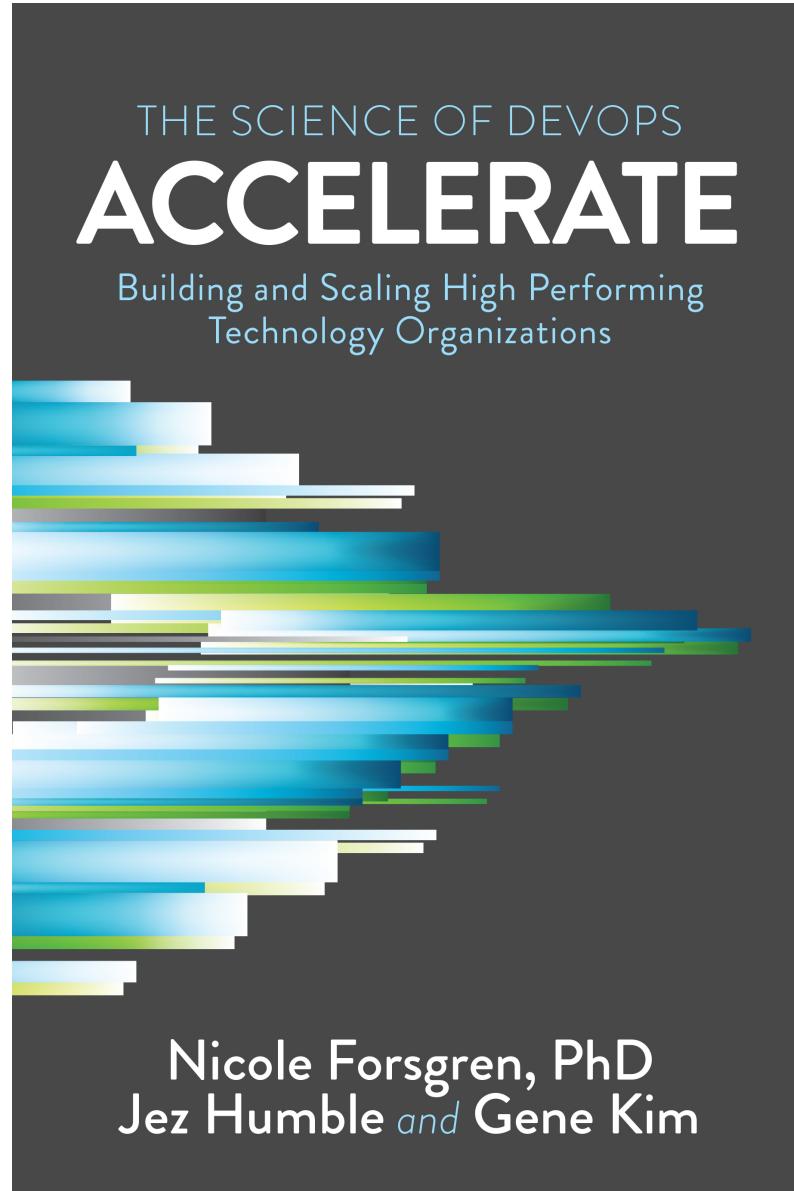
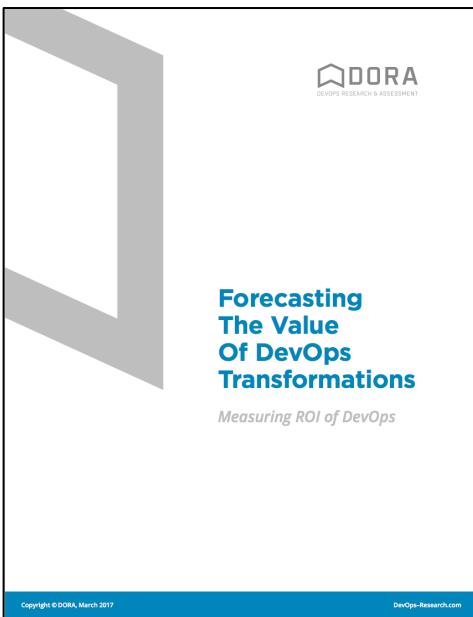
2016: Shifting left.



2017: DevOps works for all organizations.



The journey



The Highlights

- Technology matters
- Maturity models don't work
- Transformations need technology AND process AND culture
- Architecture and technology
- You can accelerate your journey
- Leadership matters
- You can help

Technology matters

“IT doesn’t matter.”

-- Nicholas Carr, 2003

DevOps is good for Technology

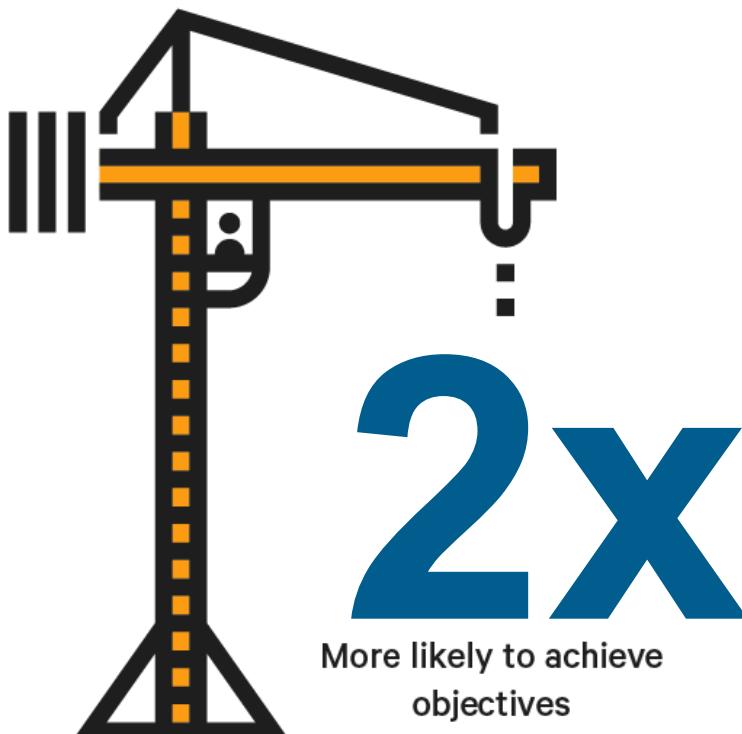
Measuring DevOps and ***Software delivery performance***

- Deploy frequency (when business demands)
- Lead Time for Changes
- Mean Time to Recover (MTTR)
- Change Fail Rate

Software Delivery Performance is
comprised of **throughput** and **stability**,
and **both are possible without tradeoffs**

**DevOps is
good for organizations**

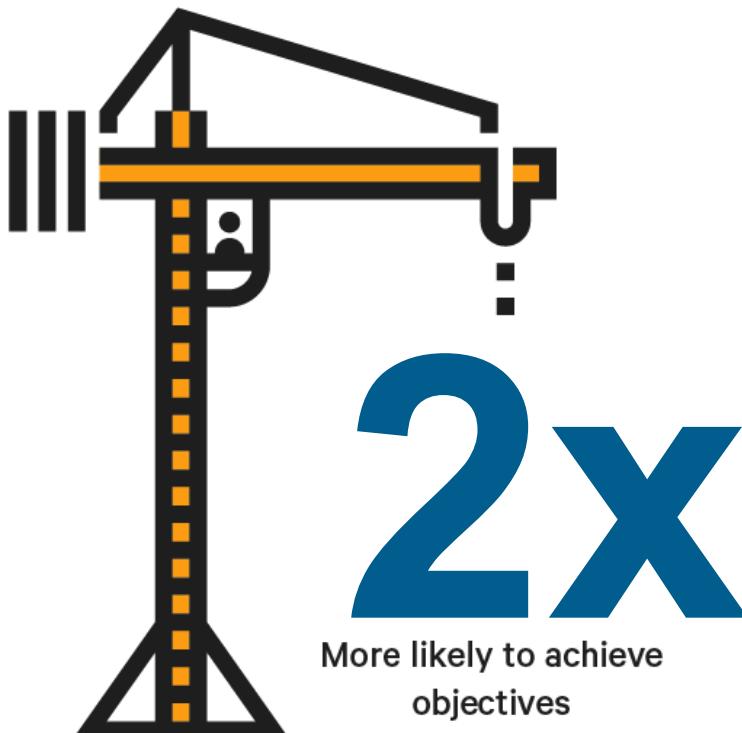
High Performing technology organizations are twice as likely to achieve or exceed



Commercial Goals

- Productivity
- Profitability
- Market Share
- # of customers

High Performing technology organizations are twice as likely to achieve or exceed



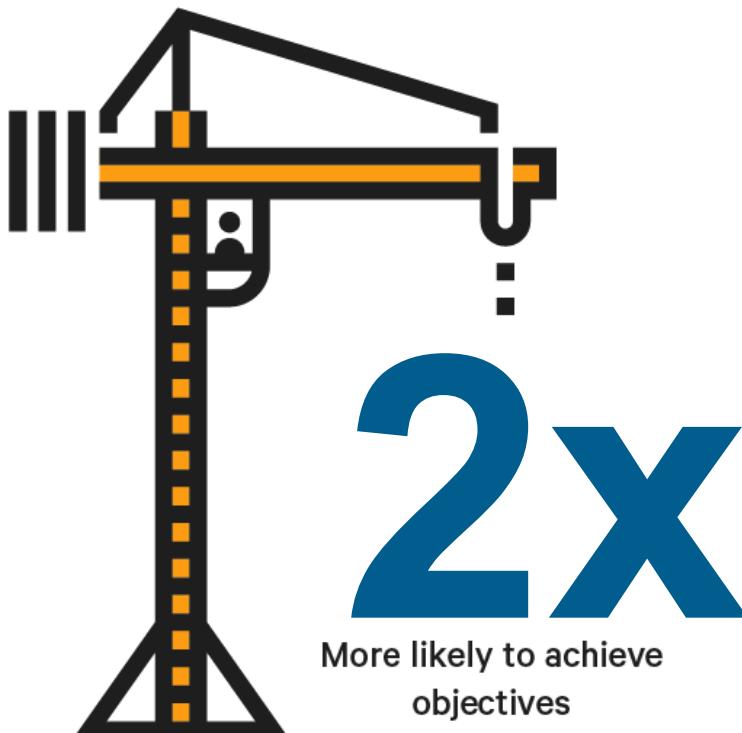
Commercial Goals

- Productivity
- Profitability
- Market Share
- # of customers

Non-commercial Goals

- Quantity of products or services
- Operating efficiency
- Customer satisfaction
- Quality of products or services
- Achieving organizational or mission goals

High Performing technology organizations are twice as likely to achieve or exceed



Commercial Goals

- Productivity
- Profitability
- Market Share
- # of customers

Non-commercial Goals

- Quantity of products or services
- Operating efficiency
- Customer satisfaction
- Quality of products or services
- Achieving organizational or mission goals

★ 50%

Higher market cap
growth over 3 years*

Maturity models don't work

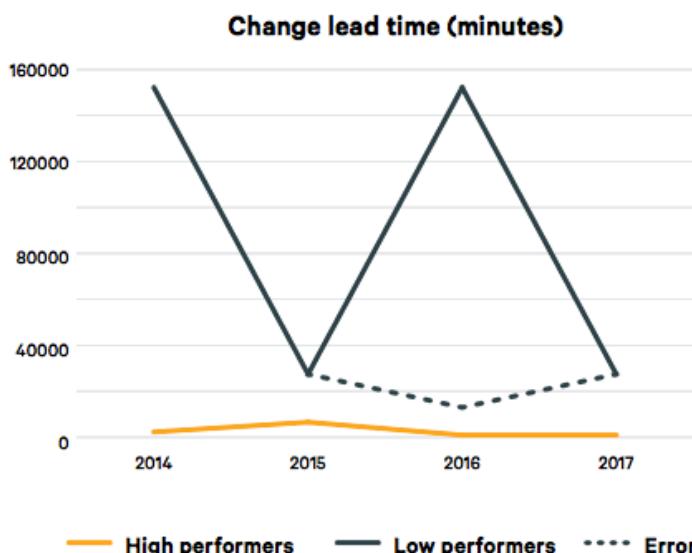
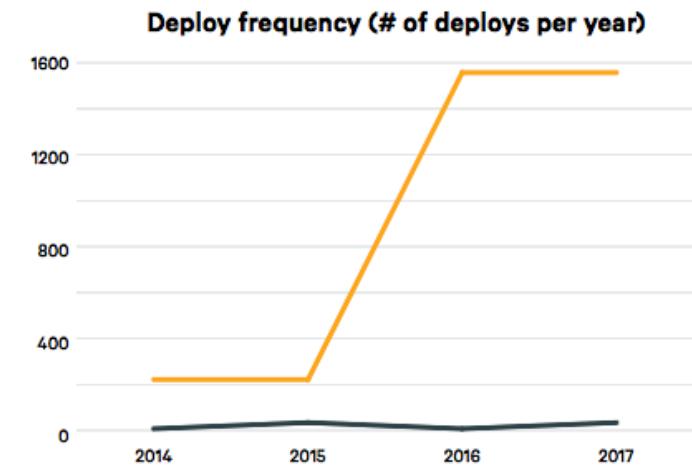


Maturity models are for **CHUMPS**



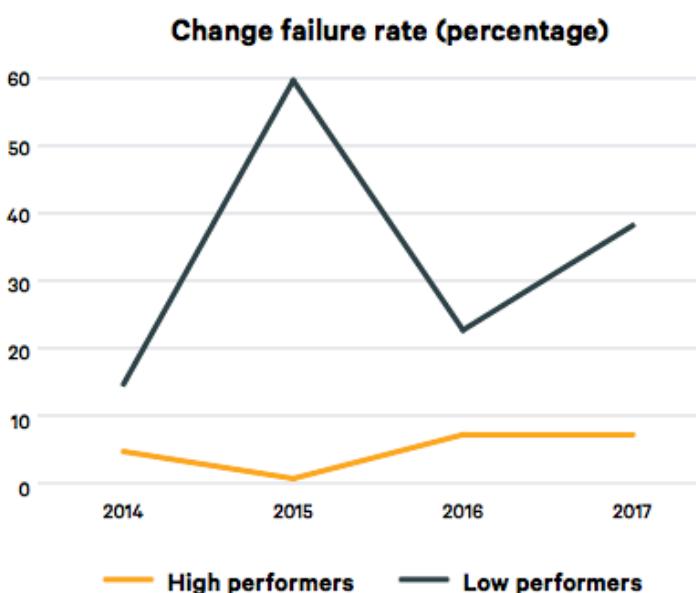
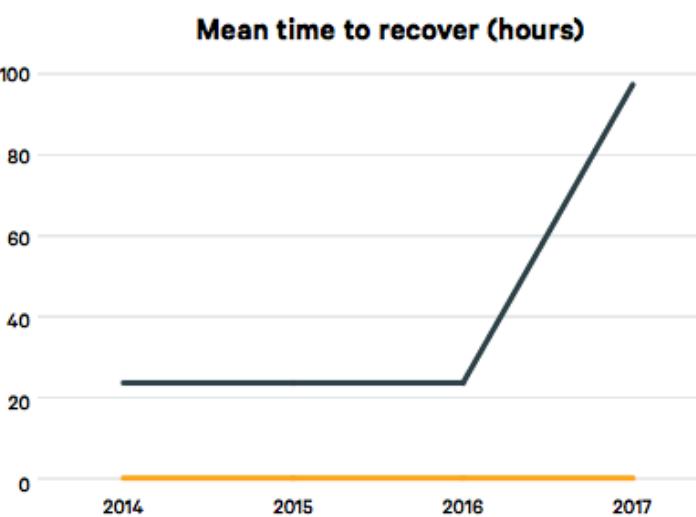
Maturity models are for **CHUMPS**

Industry changes: Year over year



* Note: Last year, our graph incorrectly reported the lead time for low performers, as shown by the dotted line.

Speed



Stability

What is good
enough in one
year is out of
date the next

High Performing DevOps teams

More *agile*

46x

More frequent
Code deployments

That's the difference between multiple times per day and once a week or less.

440x

Faster lead time from commit to deploy

That's the difference between less than an hour and more than a week.

High Performing DevOps teams

More *reliable*

96X

Faster mean time to
recover from downtime

That means high performers recover in
less than an hour instead of several days

1/5X

As likely that changes will
fail

That means high performers changes fail 0-15% of
the time, compared to 31-45% of the time.

Table 2: 2017 IT performance by cluster

| Survey questions | High IT performers | Medium IT performers | Low IT performers |
|---|---|--|---|
| Deployment frequency <i>For the primary application or service you work on, how often does your organization deploy code?</i> | On demand (multiple deploys per day) | Between once per week and once per month | Between once per week and once per month* |
| Lead time for changes <i>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 commit to code successfully running in production)?</i> | Less than one hour | Between one week and one month | Between one week and one month* |
| Mean time to recover (MTTR) <i>For the primary application or service you work on, how long does it generally take to restore service when a service incident occurs (e.g., unplanned outage, service impairment)?</i> | Less than one hour | Less than one day | Between one day and one week |
| Change failure rate <i>For the primary application or service you work on, what percentage of changes results either in degraded service or subsequently requires remediation (e.g., leads to service impairment, service outage, requires a hotfix, rollback, fix forward, patch)?</i> | 0-15% | 0-15% | 31-45% |

* Note: Low performers were lower on average (at a statistically significant level), but had the same median as the medium performers.

Transformations need
Technology AND
Process AND
Culture

DevOps is

Technical practices

seen in **Continuous Delivery**,

Management practices

seen in **Lean and Agile** principles, and

Organizational **Culture and Leadership**

Research shows that these drive Organizational Performance
and Technology Performance

“IT doesn’t matter.”

-- Nicholas Carr, 2003



@nicolefv

Remember it's “Tech Plus”

- CAMS -- the original definition, coined by Damon Edwards and John Willis at DOD Mountain View 2010
- 2014 – 2017 State of DevOps Reports
 - Technology + Lean Management + Culture
- Bessen (2017): “Real Reason Superstar Firms are Pulling Ahead”
 - Because of IT **combined** with other things, like management, brand, or IP
 - Strategic use of technology explains revenue and productivity gains more than M&A and entrepreneurship

**Architecture matters...
technology doesn't**

Technology / technology stack doesn't matter

- Low performers are more likely to:
 - be working on software developed by an outsourcing partner
 - be working on mainframe system

BUT

- Working on a mainframe system was not statistically correlated with performance.
- Working on greenfield or brownfield (or any other system) wasn't correlated with performance, either.

Architectural outcomes: can my team...

- ...make large scale changes to the design of its system without the permission of someone outside the team, or depending on other teams?
- ...complete its work without fine-grained communication and coordination with people outside the team?
- ...deploy and release its product or service on demand, independently of other services the product or service depends upon?
- ...do most of its testing on demand, without requiring an integrated test environment?
- ...perform deployments during normal business hours with negligible downtime?

Conway's Law

“organizations which design systems ...
are constrained to produce designs which
are copies of the communication
structures of these organizations”

You can *accelerate* your journey

We know there are key capabilities that drive Software Delivery Performance

- They fall into four categories:
 - Technology and automation
 - Process
 - Measurement/monitoring
 - Culture

Technology and automation

- Version control
- Deployment automation
- Continuous integration
- Trunk-based development
- Test automation
- Test data management
- Shift left on security
- Continuous delivery
- Loosely-coupled architecture
- Architect for empowered teams

Process

- Gather and implement customer feedback
- Work in small batches
- Lightweight change approval process
- Team experimentation

Trunk-based development & change approval process



By focusing on trunk-based development and streamlining their change approval processes, Capital One saw stunning improvements in just two months.

20X

Increase in Number
of Releases

30+

Number of Times Some
Applications Deploy to
Production in a Day

0

No Increase in
Incidents

Measurement and Monitoring

- Visual management
- Monitoring for business decisions
- Check system health proactively
- WIP limits
- Visualizations

Enabling Real-time Insights to Delivery Teams



@nicolefv



Culture

- Westrum organizational culture
- Climate for learning
- Collaboration among teams
- Make work meaningful
- Transformational leadership

Westrum Organizational Culture

| Pathological Power-oriented | Bureaucratic Rule-oriented | Generative Performance-oriented |
|--------------------------------|-------------------------------|------------------------------------|
| Low cooperation | Modest cooperation | High cooperation |
| Messengers shot | Messengers neglected | Messengers trained |
| Responsibilities shirked | Narrow responsibilities | Risks are shared |
| Bridging discouraged | Bridging tolerated | Bridging encouraged |
| Failure leads to scapegoating | Failure leads to justice | Failure leads to inquiry |
| Novelty crushed | Novelty leads to problems | Novelty implemented |

Where should I start?

- “It depends.” Everyone is different
- Patterns I see often:
 - Architecture is highest contributor to Continuous Delivery (SODR 2017) and shows up for very many teams (DORA: as the need for loosely-coupled architecture or trunk-based development)
 - Lightweight change approval process is a constraint for most teams (DORA)
 - Continuous integration (DORA – and its full complement)

So what to do?

1. Identify your constraints. Pick “a few.”
2. Work to eliminate those constraints.
3. Re-evaluate your environment and system.
4. Rinse and repeat.

Leadership matters

Dimensions of transformational leadership

Vision

- Understands organizational direction.
- Understands team direction.
- Understands 5-year horizon for team.

Personal recognition

- Commends team for better-than-average work.
- Acknowledges improvement in quality of work.
- Personally compliments individuals' outstanding work.



Intellectual stimulation

- Challenges team status quo.
- Challenges team to constantly ask new questions.
- Challenges team on basic assumptions about the work.

Inspirational communication

- Inspires pride in being part of the team.
- Says positive things about the team.
- Inspires passion and motivation; encourages people to see that change brings opportunities.

Supportive leadership

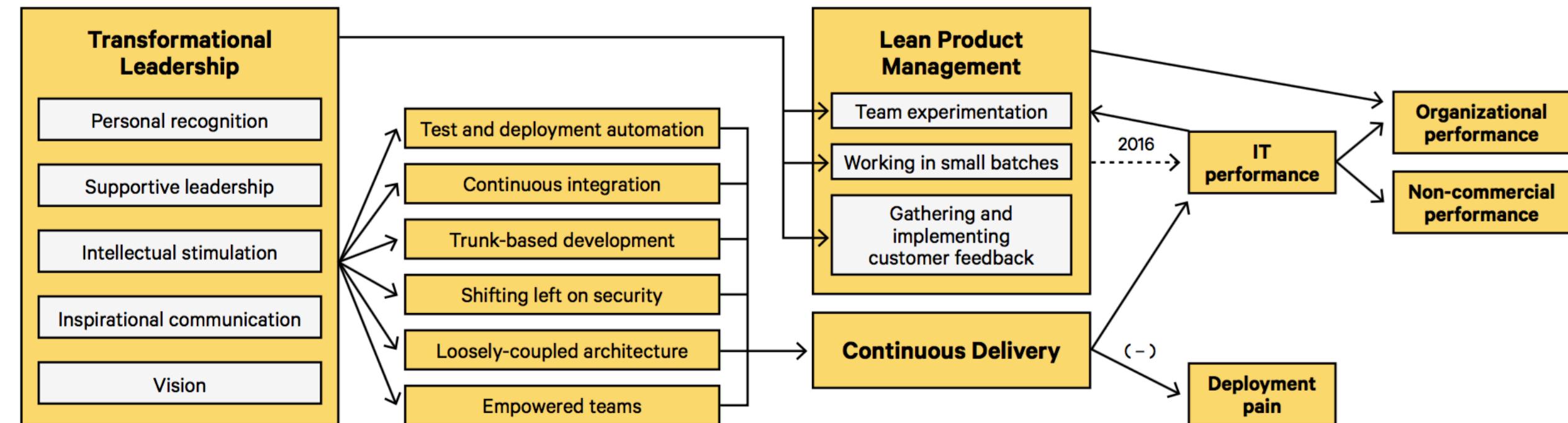
- Considers others' personal feelings before acting.
- Is thoughtful of others' personal needs.
- Cares about individuals' interests.

Leadership Necessary but not Sufficient

- Teams with the least transformational leaders (the bottom third) were one-half as likely to be high IT performers
- Leaders cannot do it alone! Teams with the top 10% of transformational leaders performed no better than the median

Relationship between transformational leadership and performance

Figure 1. Structured equation model showing relationships between constructs

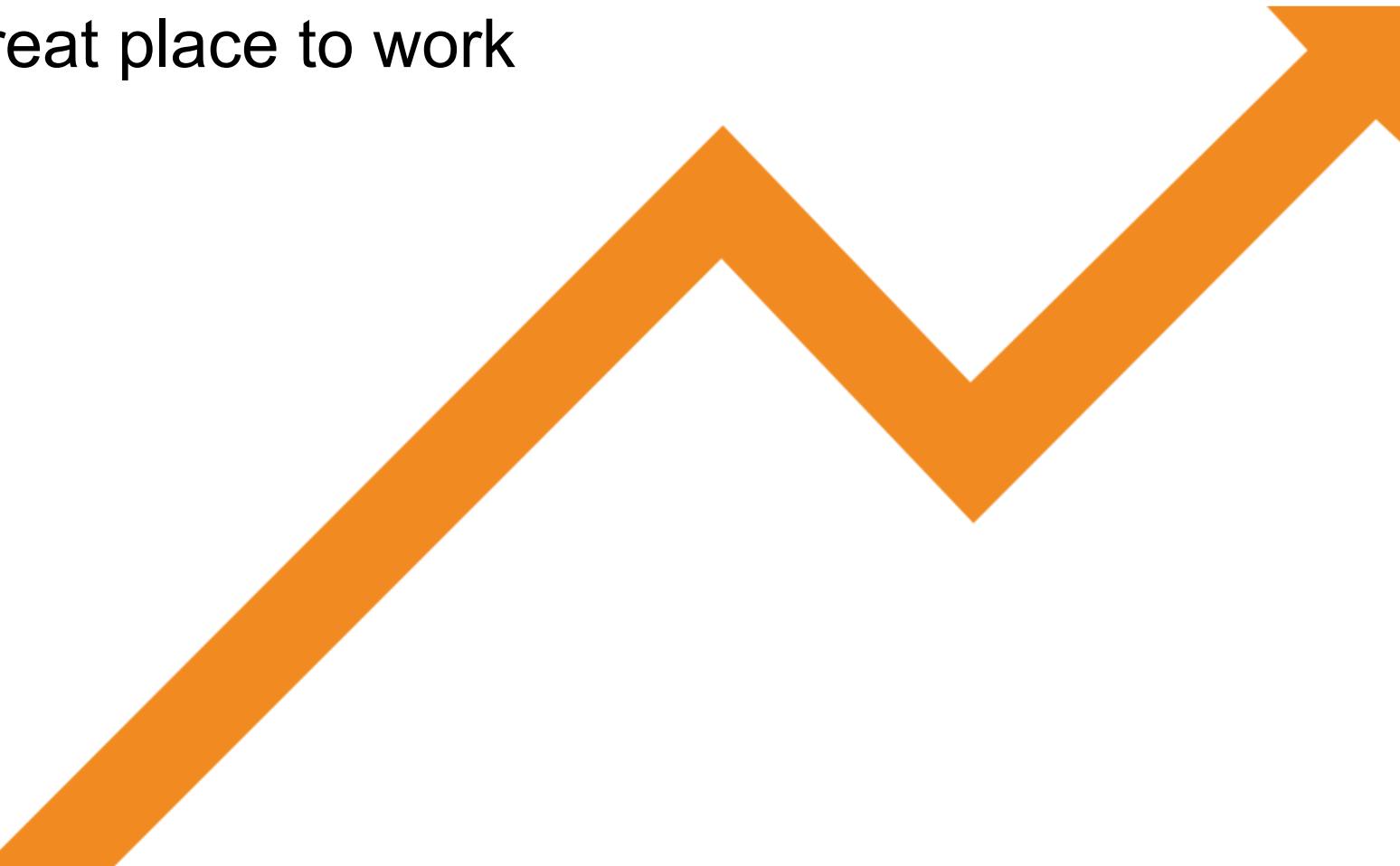


Good practices and smart investment make the work better, too!

- The **work**?
 - Less deployment pain
 - Less burnout
 - Higher employee Net Promoter Score

Employees in high performing organizations are 2.2 times more likely to recommend their organization as a great place to work

2.2x



You can help

Your role in this

- Be the transformational leaders. Own this.
- Measure five things
 - Focus on outcomes: Software delivery performance (speed & stability)
 - Drive performance improvements – both with tech and with *not* tech
- Share your stories!

For more information:



For our ROI whitepaper,
case studies, the State of
DevOps Reports & peer-
reviewed research, visit
devops-research.com

Thank you

nicole@devops-research.com
[@nicolefv](https://twitter.com/nicolefv)
nicolefv.com
devops-research.com

High Performing DevOps teams

More *agile*

46x

More frequent
deployments

440x

Faster lead times

What does this mean for:

New content delivery

Value/savings around A/B testing

Value around speed to market

Compliance / regulatory

Security

Evaluating well-designed and executed experiments that were designed to improve a key metric, **only about 1/3** were successful at improving the key metric!

High Performing DevOps teams

More *reliable*

What does this mean for:

5x

Fewer deploy
failures

Value/savings around reliability

Value/savings around uptime

Compliance

Security

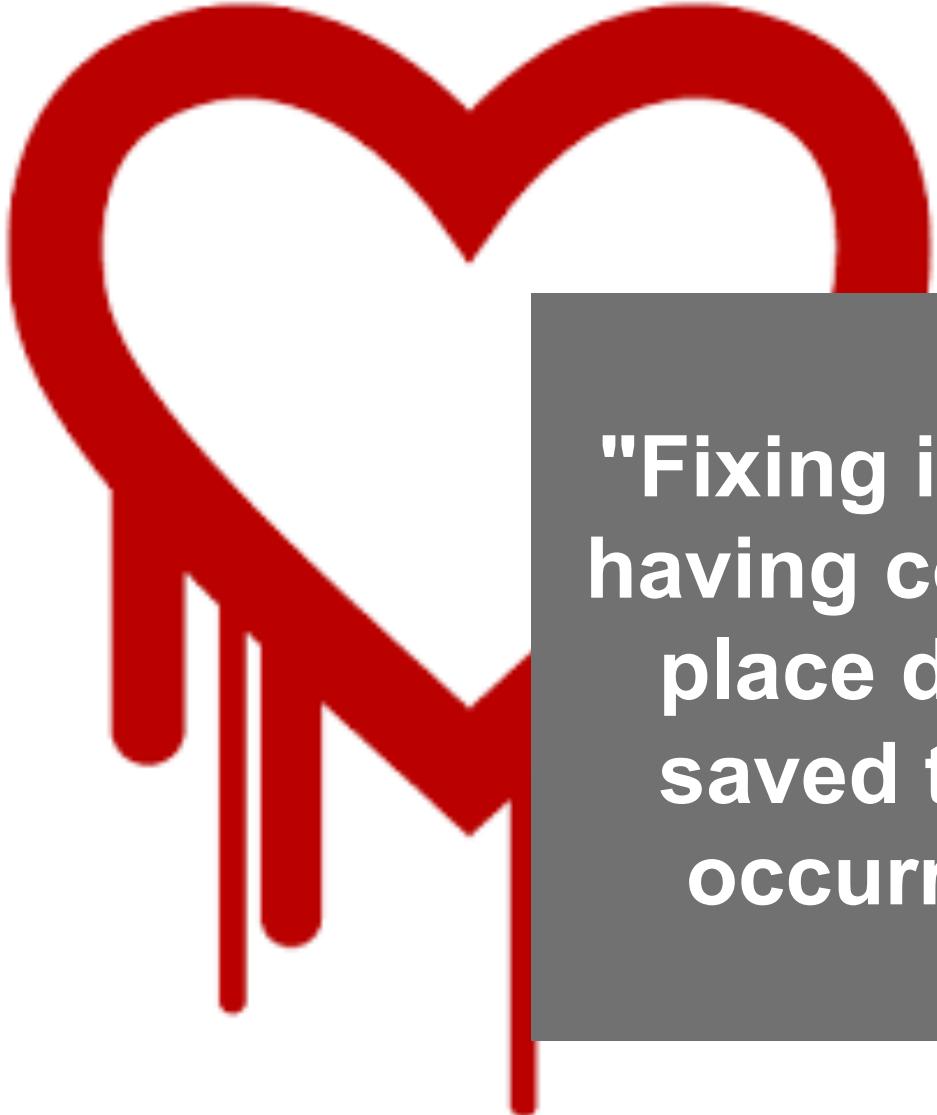
Reputation around uptime,
compliance & security

96x

Faster MTTR

NETFLIX





"Fixing it as soon as possible or having compensating controls in place days before could have saved this entire breach from occurring in the first place."



Continuous Delivery



Continuous Delivery

- Version control
- Test data management
- Test automation
- Deployment automation
- Trunk-based development
- Cont integration
- Security
- Architecture
- Empowered teams



“We never had testability before. We have it now. We have this experience and know this stuff is working, and working with controls.”

– Product Owner for Yahoo Chef implementation



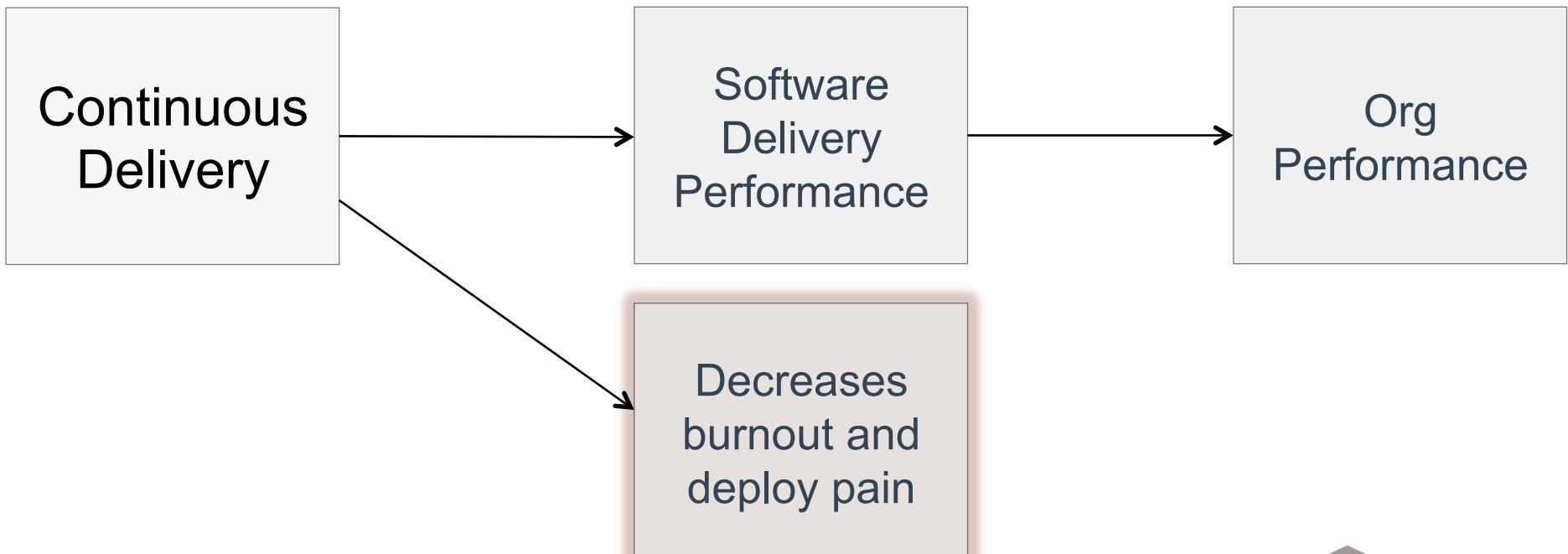
Automated configuration and deployment of 250,000 nodes

Can deploy up to 140k node configurations in 8 hours.

Can patch entire infrastructure within 6 hours of a patch being made available

Continuous Delivery also makes our work *feel* better

- Version control
- Test data management
- Test automation
- Deployment automation
- Trunk-based development
- Cont integration
- Security
- Architecture
- Empowered teams



Microsoft Engineering: DevOps Lessons

Thiago Almeida -- DevOps Days London, 2016



Work/Life Scores

Before CD

38%

After CD:

75%

Source: <https://vimeo.com/165184757>

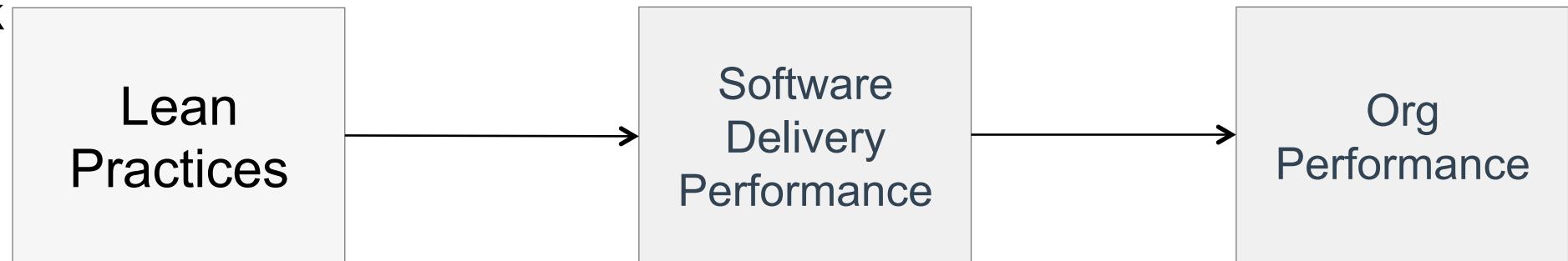
But what else **drives** Performance?

Lean Management Practices

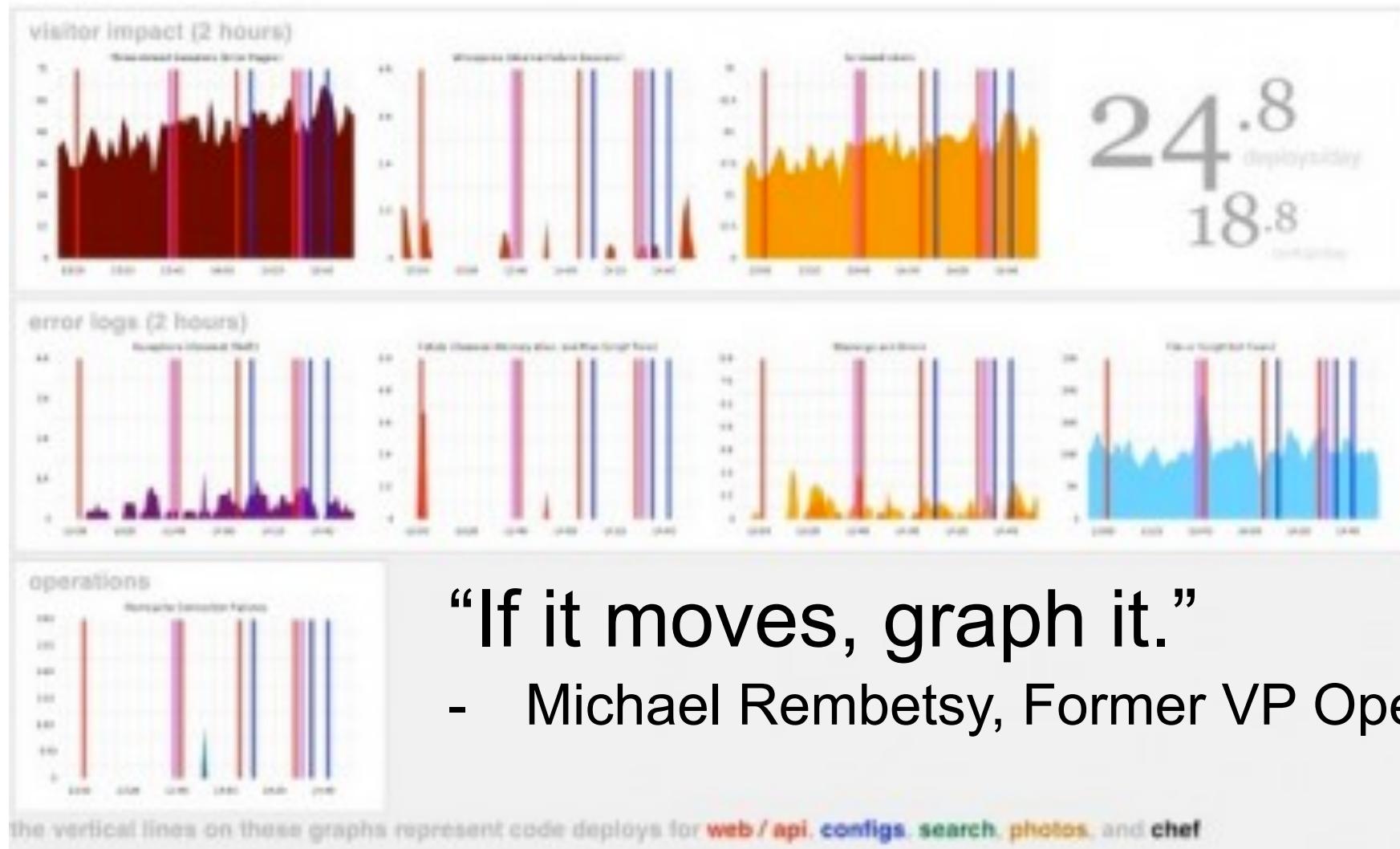


Lean Management Practices

- WIP limits
- Visual displays
- Monitoring tools
- Integrating customer feedback
- Small batches
- Visual management
- Lightweight change approvals
- Team experimentation



Visual Displays



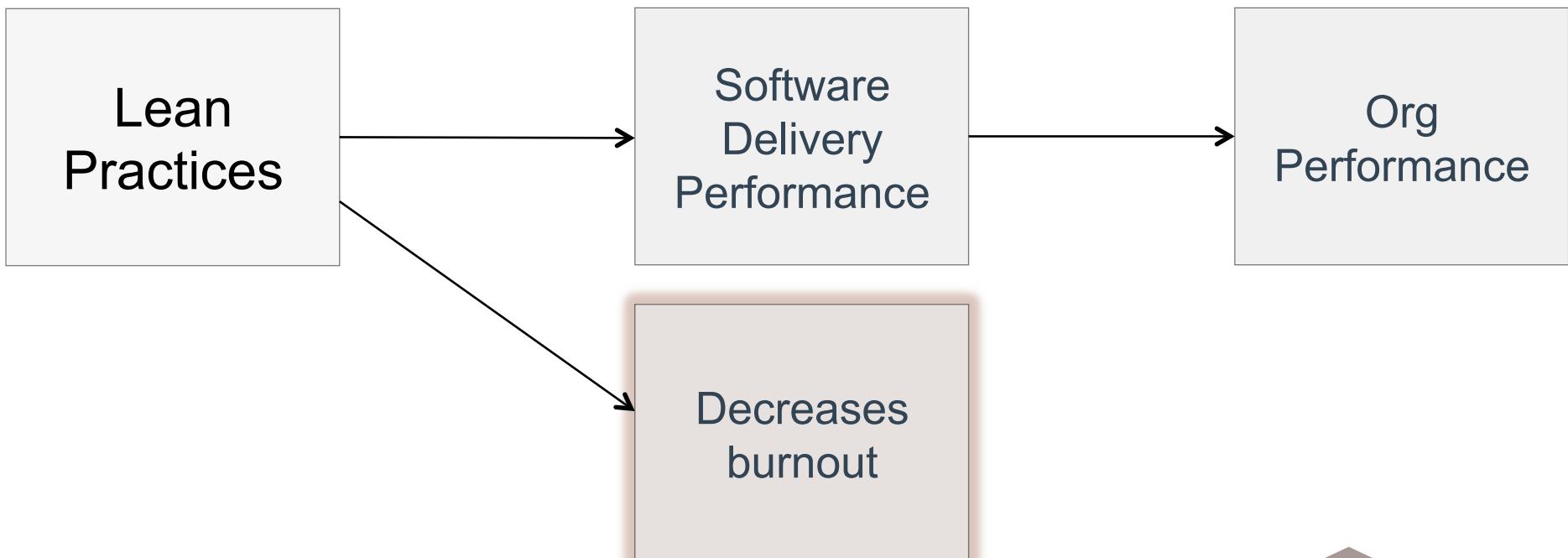
“If it moves, graph it.”

- Michael Rembetsy, Former VP Operations, Etsy



Lean Management Practices also make work *feel* better

- WIP limits
- Visual displays
- Monitoring tools
- Integrating customer feedback
- Small batches
- Visual management
- Lightweight change approvals
- Team experimentation



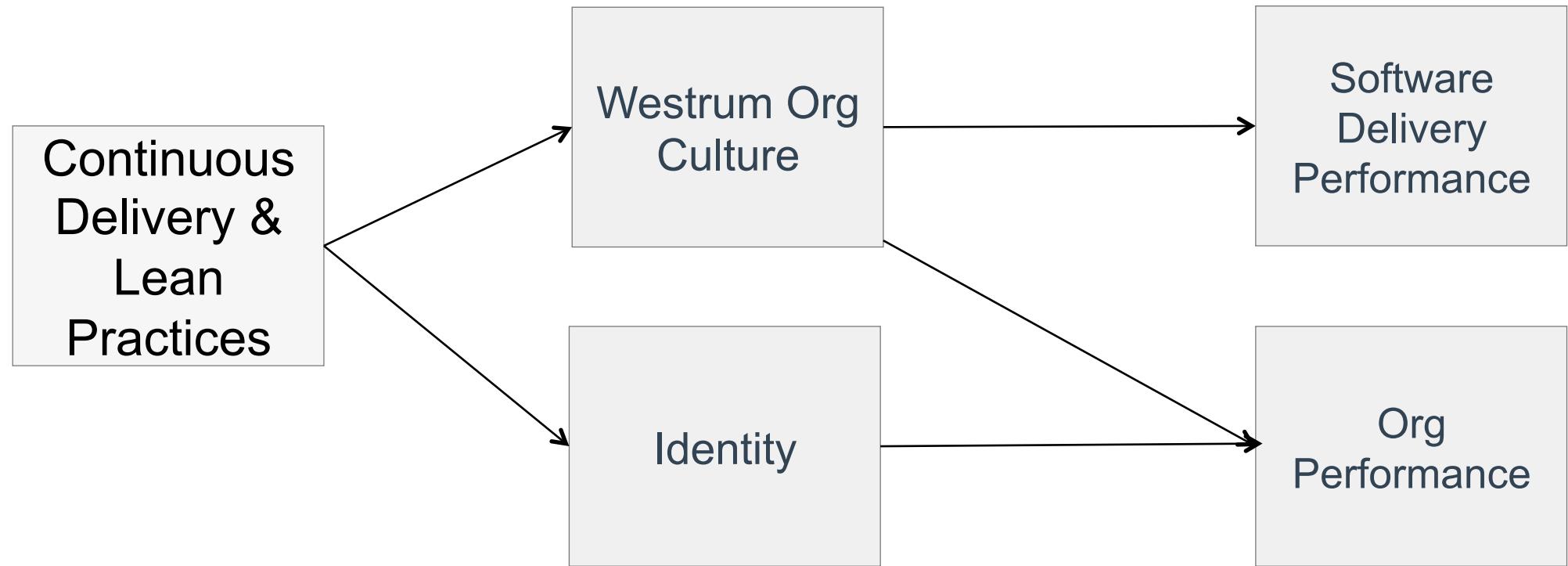
“I was trying to figure out why my team was working themselves to death but not getting anything done... By implementing WIP limits, we were able to focus on our work. Finishing work feels better than sprinting and feeling like a hero in the moment, because that's only a moment.”



- Julia Wester,
Development Manager
for Turner Sports,
Turner Broadcasting

But wait... there's more

CD and Lean Management practices also improve our **culture** and **identity**, which improve SW Delivery & Org Performance



Westrum Organizational Culture

| Pathological Power-oriented | Bureaucratic Rule-oriented | Generative Performance-oriented |
|--------------------------------|-------------------------------|------------------------------------|
| Low cooperation | Modest cooperation | High cooperation |
| Messengers shot | Messengers neglected | Messengers trained |
| Responsibilities shirked | Narrow responsibilities | Risks are shared |
| Bridging discouraged | Bridging tolerated | Bridging encouraged |
| Failure leads to scapegoating | Failure leads to justice | Failure leads to inquiry |
| Novelty crushed | Novelty leads to problems | Novelty implemented |

Identity

- I am glad I chose to work for this organization rather than another company.
- I talk of this organization to my friends as a great company to work for.
- I am willing to put in a great deal of effort beyond what is normally expected to help my organization to be successful.
- I find that my values and my organization's values are very similar.
- In general, the people employed by my organization are working toward the same goal.
- I feel that my organization cares about me.

Adapted from Atreyi Kankanhalli, Bernard C.Y. Tan, and Kwok-Kee Wei (2005), “Contributing Knowledge to Electronic Knowledge Repositories: An Empirical Investigation,” MIS Quarterly, 29, 113-143.

Google Team Performance

Surprise! No magic formula for what makes the **perfect** team



Intuit

“By installing a rampant ***innovation culture***, we performed 165 experiments in the peak three months of tax season.

Our business result? ***Conversion rate*** of the website is up 50%. ***Employee result?*** ***Everyone loves it***, because their new ideas can make it to market.”

- Scott Cook, Intuit founder

Technology *Does* Matter

- Times – **and IT (technology)** – have changed
- DevOps is good for **Organizations**
- DevOps is good for **Technology**
- And then some detail: **What drives this change?**
 - Technical practices (hint: Continuous Delivery)
 - Management practices (hint: from Lean Management)
 - Culture
- DevOps principles can be applied throughout the business
to accelerate value