

Morgan Stanley



# Automated Change Management

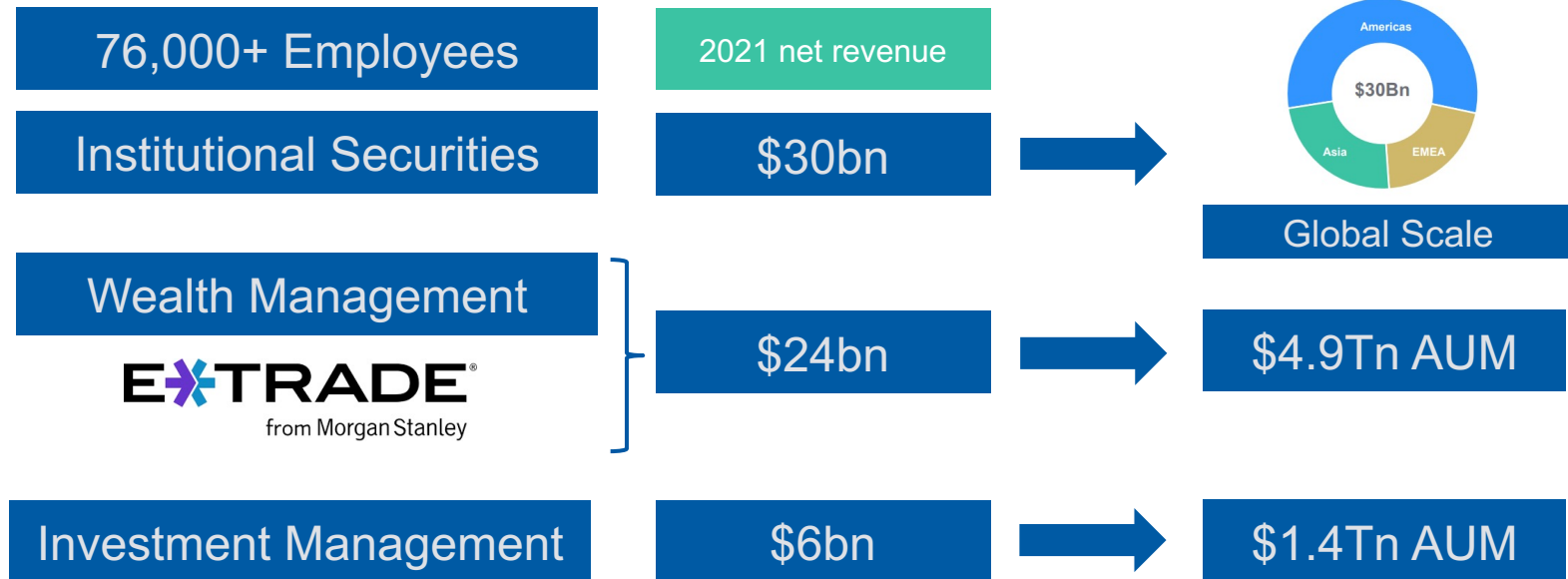
**Gus Paul**

**Product Owner, Software Delivery Assurance**

# Morgan Stanley

## Global Financial Services Firm – Three Major Business Areas

<https://www.morganstanley.com/about-us/>



Data from 4Q2021 Strategic Update <https://www.morganstanley.com/content/dam/msdotcom/en/about-us-ir/shareholder/4q2021-strategic-update.pdf>

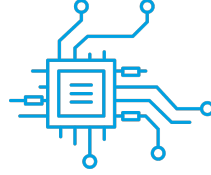
# Morgan Stanley Technology

<https://www.morganstanley.com/about-us/technology>



15K+

Technologists Worldwide



3.5K+

Applications/Systems



10B

Transactions Supported  
Annually

- Long heritage of cutting edge home grown platforms e.g. server management, binary artifact distribution, market data distribution.
- Growing every year + became a US bank holding company in 2008.
  - Increased oversight and formalisation of processes.
- Formal Agile & Cloud transformation begun in 2018.
  - DevOps recognised as a critical enabler for both.

## DevOps at Morgan Stanley

- 2018/2019 Internal knowledge sharing, metrics development (based on DORA metrics), leveraging our existing centralised CI/CD pipeline to increase automated build & deployment.
- 2019-2022+ Strategic DevOps Program with three key areas.



**Accelerate  
Software  
Development  
and Delivery**



**Increase  
Predictability,  
Frequency and  
Quality of  
Change**



**Revolutionize  
how we operate  
technology**

## Why Change Management?






*“We investigated further the case of approval by an external body to see if this practice correlated with stability. We found that external approvals were negatively correlated with lead time, deployment frequency, and restore time, and had no correlation with change fail rate. In short, approval by an external body (such as a manager or CAB) simply doesn’t work to increase the stability of production systems, measured by the time to restore service and change fail rate. However, it certainly slows things down. It is, in fact, worse than having no change approval process at all.”*

*Accelerate, Chapter 7*

# Replacing our 'one size fits all' process for Change

## Why do we need to change the way we manage Change?

	<b>Manual, Outdated Processes</b>	<ul style="list-style-type: none"><li>• Designed for Infrequent, big bang Change delivered to a less complex technology landscape</li><li>• Multiple approvals</li><li>• Manually created Change records</li><li>• Change Management invoked after the Software Delivery Lifecycle</li></ul>
	<b>Unreliable Risk Assessments</b>	<ul style="list-style-type: none"><li>• Human led risk assessments, limited by human bias and knowledge.</li><li>• Most unique Changes inherit that risk rating from a previous Change or Change template</li></ul>
	<b>Misaligned to Modern Approaches</b>	<ul style="list-style-type: none"><li>• Processes misaligned to modern approaches (e.g. ITIL v4)</li><li>• We need to maximize business benefits and manage risk from Change whilst operating productively and transparently</li><li>• Change is seen as a barrier not an enabler</li></ul>

## Addressable Opportunity: Software Change Management

Year	Production changes	Production systems	Software changes	Software change as % of all change	Software Systems
2019	160,216	3,614	89,783	56%	2,607
2020	156,689	3,523	93,291	59%	2,621
2021	175,142	3,513	110,917	63%	2,656

- We estimate between **200,000 hours** in manpower is spent a year on the current workflow – just preparing and approving change tickets.
- This process also encourages batching of releases and releasing at inconvenient times
- Software change volume is growing each year, and growing as a proportion of all change

**These numbers are expected to grow with increased frequency of deployment and increased use of e.g. infrastructure as code and on public cloud**

## SDLC & Change Management

- Software deployment is split between release preparation (SDLC) and release deployment (CM)
- SDLC Before Renovation
  - No credit for activities in dev cycle, 3 distinct senior approvals required on SDLC Release Artifact



- SDLC After Renovation



- Credit for requirements, code review, automated testing activities that were already happening
- Implied release level approval - maintains separation of duties



Squad level manual approval



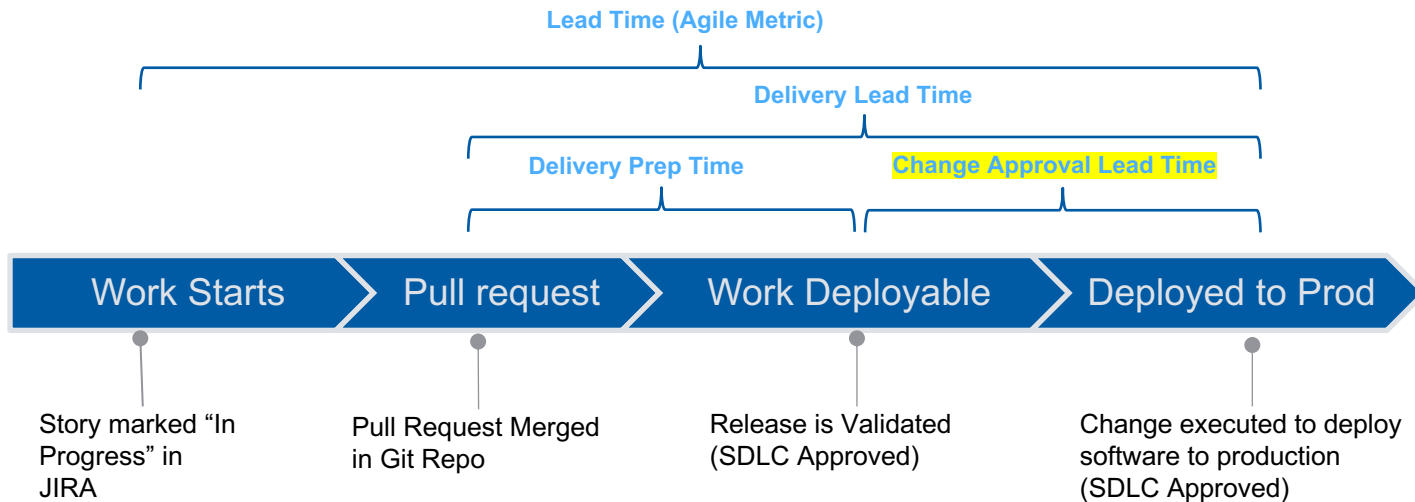
Automated output



Senior level manual approval



## Addressable Opportunity: Delivery Lead Times, Release Size



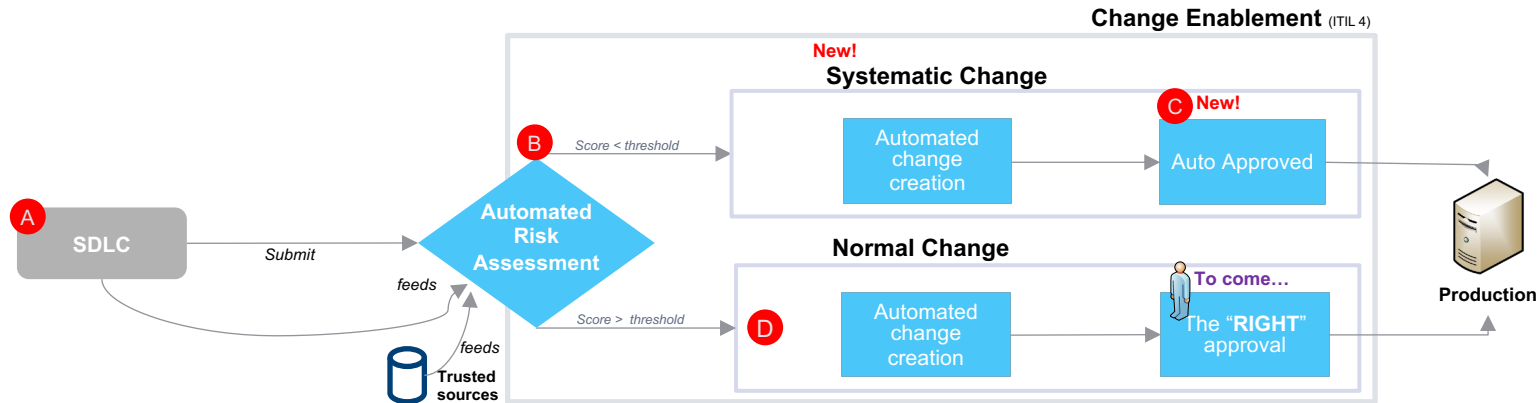
- Change Approval Lead Time 3.5 days on average
  - Some changes can have up to 7 distinct human approvers
- Post Approval Lead Time ~0.5 day
  - Some due to system architecture but also batching
  - Bigger releases at weekend than weekdays; More incident impact from weekend deployments

# Automated Change Management: The Future

Focus on enabling Change (a path, rather than a gate/barrier), whilst meeting regulatory requirements

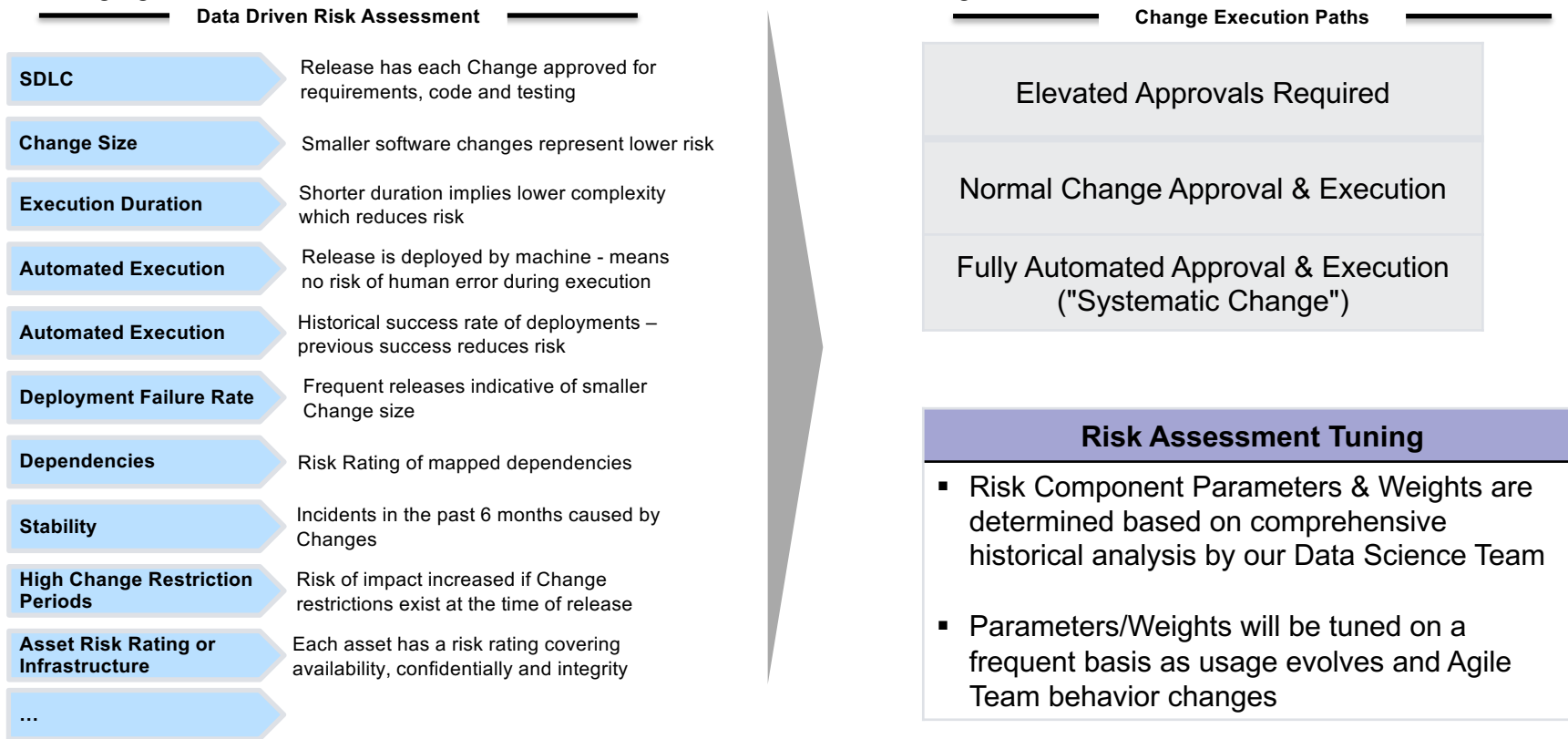
Systematic Change: Fully automated Change Record throughout the entire Change Life Cycle

- A. SDLC focus in reducing deployment risks during the software development cycle
- B. Risk assessed automatically from data points from trusted sources (*rather than subjectively by humans*)
- C. Change is auto approved (*approvals inherited from SDLC*)
- D. If not eligible for Systematic Change, the same risk assessment will be used in the Normal Change route

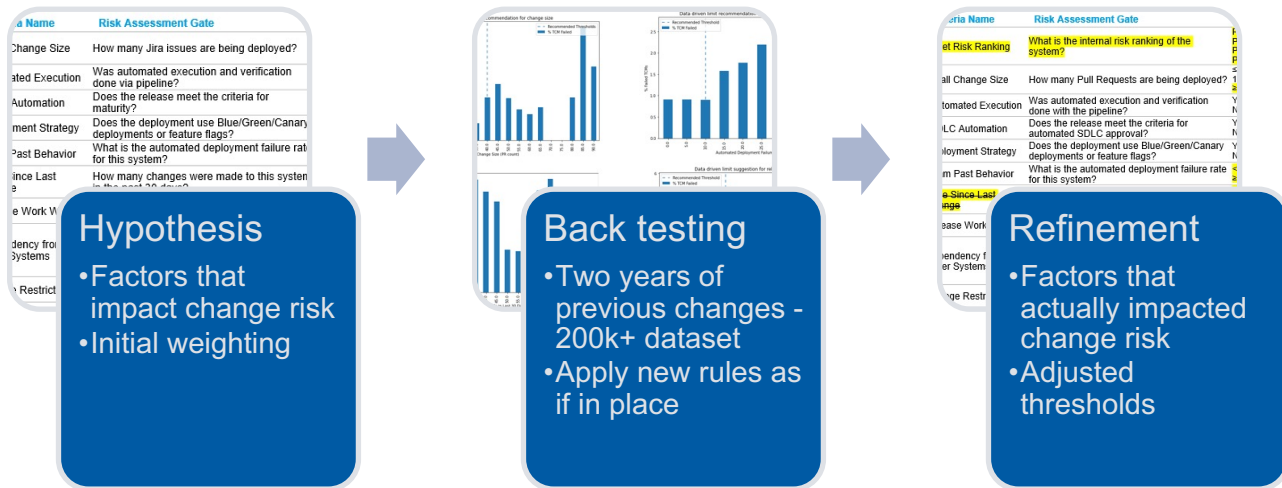


# Data Driven Risk Assessment

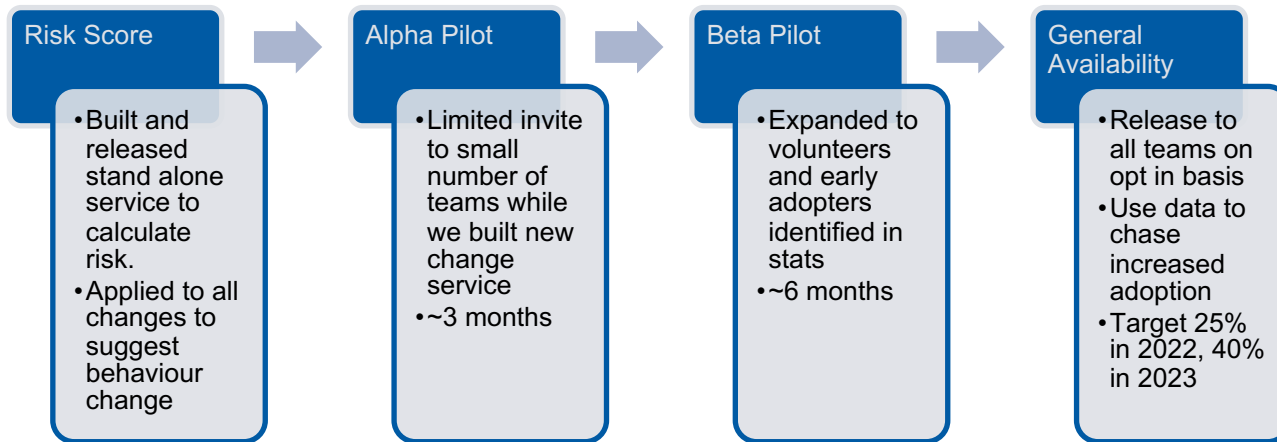
Leveraging automation to produce quantitative risk assessments for Change



# Building and testing the risk criteria



## How we built and rolled out



➤ Beta Pilot: Increase Confidence; overcome resistance

- 1,556 changes across 58 Systems.
- Zero change related incidents.
- 78% improved delivery lead time
- \$10k approvals saving

# Systematic Change: Case Studies

## Not just about Speed, it's also about reducing risk

### "Bug identified at 17:57.... Fix on Prod by 18:42!!"

- Bug identified (by Product Owner) in feature at 17:57
- Fix identified and code & unit test updates committed by 18:24
- Automated SDLC began release preparation at 18:25
- Systematic Change ticket auto Approved at 18:33
- Automated Execution begins at 18:39, completed by 18:42
- Speed:
  1. This same product previously had a 2-week deployment cycle, leading to long lead times for changes
  2. Issue fixed without invoking *any* exception process (e.g., no break glass access to Production, no emergency change)

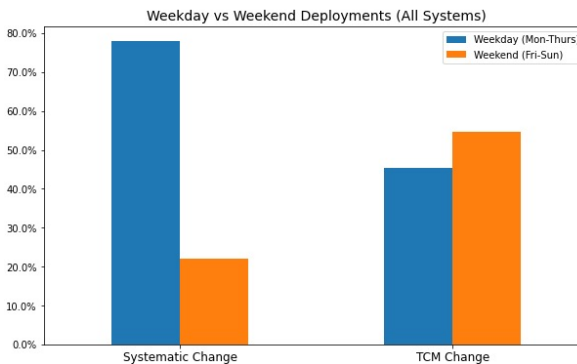
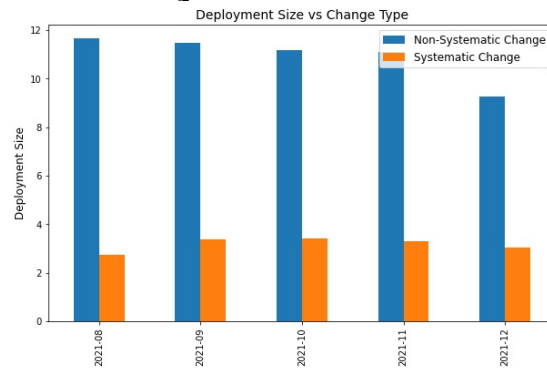
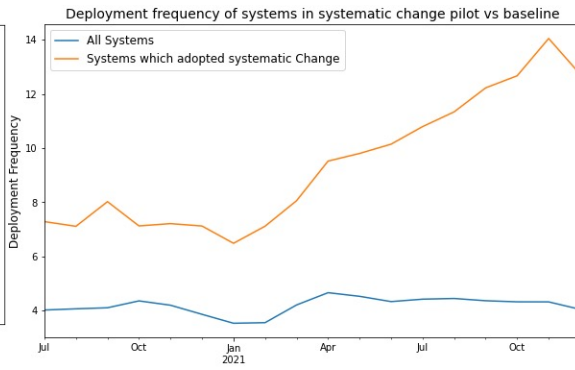
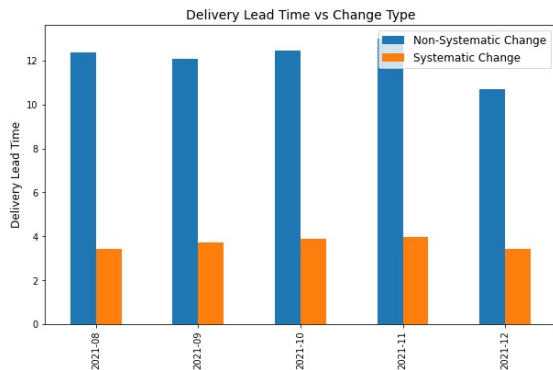
### "Systematic Change to the rescue"

- A deployment (legacy change) for internal tool resulted in the scheduled job for onboarding of requests to fail.
- Using Systematic Change, the team was able to create a fix and deploy it to production before it became a larger issue
- Risk Reduction:
  1. Speed / automated nature of Systematic Change enabled team to quickly issue a permanent fix
  2. Delivery followed all standard processes and best practices applied ( Fully Automated Deploy & Verify)
  3. Issue fixed without invoking *any* exception process (e.g., no break glass access to Production, no emergency changes)

# Data Insights: Impact of Systematic Change on Delivery KPIs

Analysis of Systematic Changes vs regular changes shows:

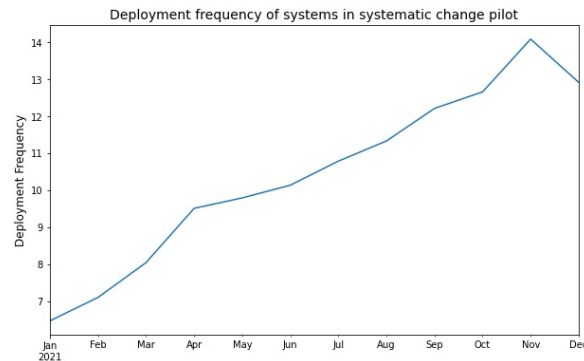
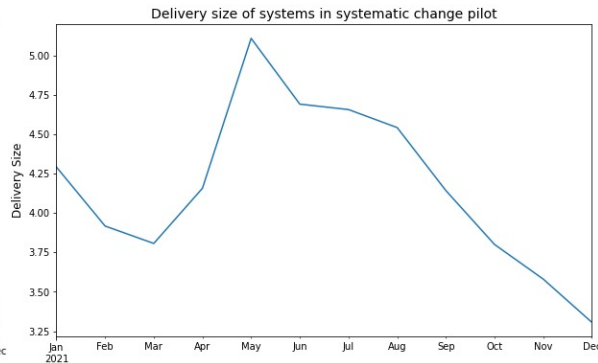
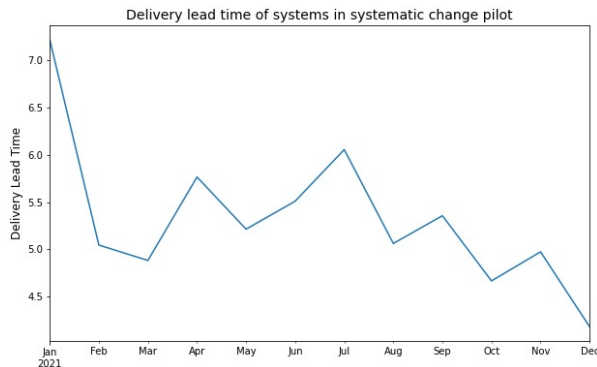
- **Lower** Delivery Lead Time
- **Significantly Increased** Deployment Frequency
- **Lower** Delivery Size (*known to be associated with reduced risk*)
- Higher proportion of **weekday** deployments



## Data Insights: Impact of Systematic Change on Delivery KPIs

Majority of systems participating in Systematic Change pilot show progress on main DevOps KPIs:

- Reduction of Delivery Lead Time (in 75% of systems)
- Reduction of Delivery Size (in 90% of systems)
- Increase of Deployment Frequency (in 77% of systems)





## Testimonials on Automated Change Management

“...it really has been a difference. We've been opportunistic, experimenting with and becoming early adopters of new capabilities and moved towards a more sustainable way of working (see an issue, fix the issue, ship the fix).”

*Distinguished Engineer, Identity Management*

“I am so happy for this...I truly believe this is one of the most pivotal changes I have ever been a part of in my 20 years here!”

*Executive Director, Product Owner*

“Problem today, fix today, production tomorrow - it feels like I'm working for a different company.”

*VP, Senior Engineer*

“Weekday rocks. I could schedule the deploy when my workday begins and I'll have plenty of time to verify, triple check and believe that it actually worked.”

*VP, Senior Engineer*

“The transition feels like a chapter straight out of The Phoenix Project”

*Executive Director, Tech Lead*

## Testimonials on the DevOps program

“We were concerned that with a weekly deployment cycle, we would see a dip in quality, but ....we significantly decreased the risk of a release since we deployed smaller, more frequent releases.”

*Managing Director, Wealth Management*

“Through the adoption of DevOps, the team has been able to release daily and iterate rapidly, deploying more frequently while lowering risk. That’s a major benefit considering the Electronic Trading plant is one of the main revenue drivers for the division, where technology is key and being able to react quickly makes the difference between winning a client’s business or not.”

*Managing Director, Algorithmic Trading*

“Thanks to DevOps, the team was able to combine speed of delivery with high quality work, performing daily deployments of the new IPO portal as they rapidly iterated with the teams from Investment Banking and Global Capital Markets. It enabled this portal to be rapidly built and deployed flawlessly for the largest technology IPO of 2020.”

*Managing Director, Technology Investment Banking.*

## Help we are looking for?

- We want to collaborate with other regulated firms over a consistent view to regulators over benefits of automation



<https://github.com/finos/devops-mutualization>

- How to we further improve the automated risk assessment
  - Particularly what data points should we consider for infrastructure/hardware changes
  - We are looking at vendors with AI/ML Models
- Best practices for demonstrating all of this in a simple to follow audit trail
  - We have linkages through the different tools, but it would be easier for audit to see it all in one place