Agile + DevOps EAST

A TECHWELL EVENT

DT4

DevOps Leadership
Thursday, November 8th, 2018 11:30 AM

Dominating DevOps with Distributed Teams

Presented by:

Treasa Overton

Overton Coaching
‘

Brought to you by:

TECHWELL™

350 Corporate Way, Suite 400, Orange Park, FL 32073
888-268-8770 ‥ 904-278-0524 - info@techwell.com - http://www.starwest.techwell.com/

Treasa Overton

Treasa has been a consultant for twelve years, recently taking on the role of lead agile coach at USAA in their transformation efforts. With more than twenty years' experience in the development of teams, her career has evolved into being a Certified Org Mindset Enterprise Coach. Working with Fortune 100 to Fortune 1000 companies, she has trained more than a thousand people on how to work with distributed teams and why implementing DevOps practices into team norms is a must to stay viable in delivering business value. Both onsite and remotely, she's trained hundreds of leaders, change agents, ScrumMasters, product managers, product owners, and teams.

Dominate DevOps with Distributed Teams

November 4-9

Agile + DevOps 2018

Treasa Overton is a Certified Professional Transformation Coach through the ICF, she is also holds certifications in SAFe as an SPC4, Certified AgilityHealth Facilitator and a Certified Scrum Professional.

Treasa has trained more than a thousand Leaders, Product Owners and Scrum Masters in both SAFe and customized courses.

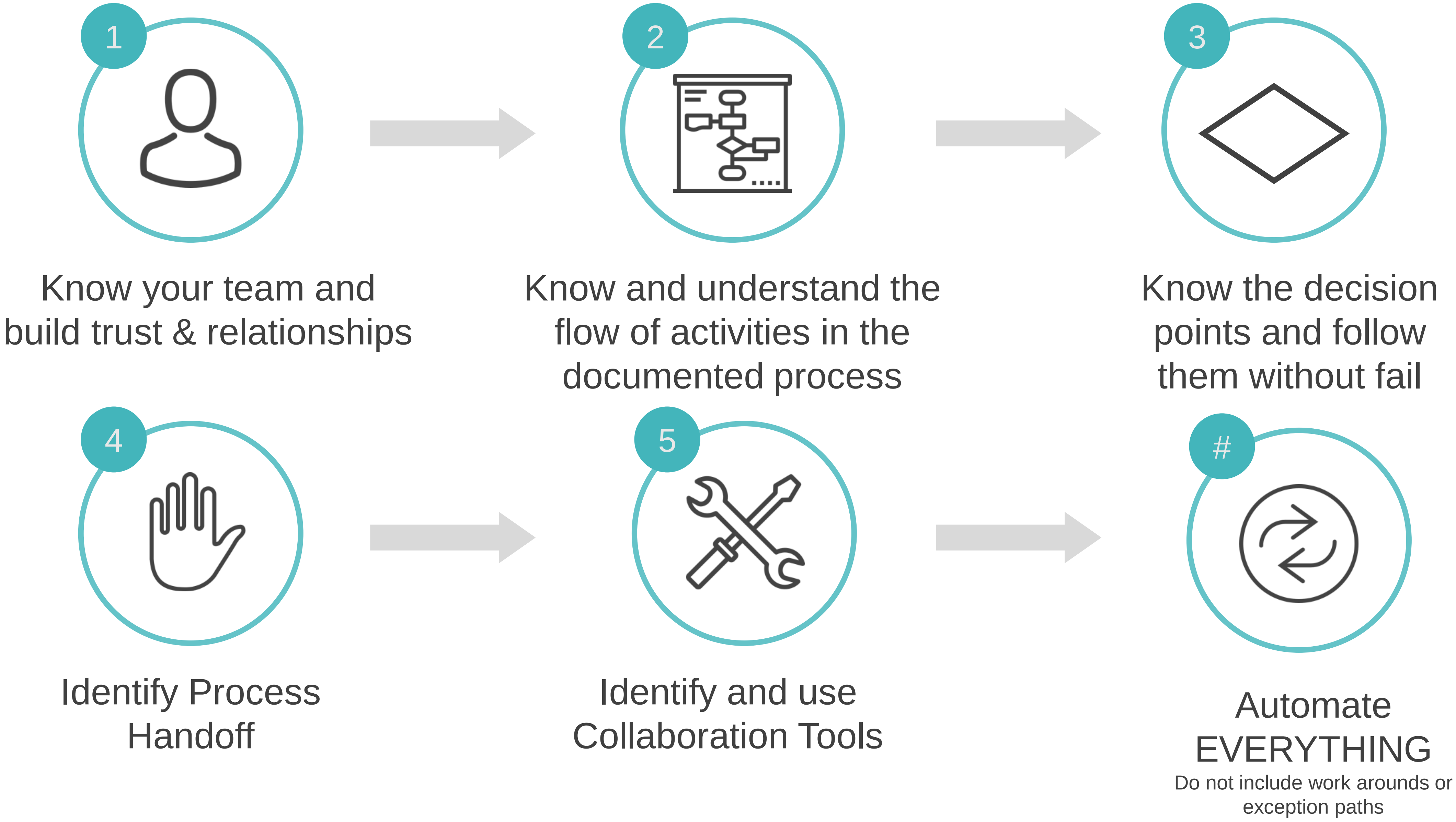
Treasa has 12 years of experience as an Agile Coach and has primarily spent her career working in every role of an Agile Team for mostly larger Fortune 100 and 500 companies, guiding them in their agile transformations and personal growth journeys.

Treasa mentors other Agile Coaches as well receives their coaching feeling that its critical to embrace the continuous learner mentality. Coaching Scrum Masters to rise to the challenge of becoming team coaches is a passion of hers as well as building strong communities of practice to never stop growing.

As a military spouse Treasa is thrilled that she has a skill that she can use to give back to some of our service members by volunteering to coach veterans transitioning out of the military to the civilian workforce.

Treasa’s personal motto is “If your not having fun, then you’re doing it wrong!!!

Agenda



1 Know your team and build trust & relationships

2 Know and understand the flow of activities in the documented process

3 Know the decision points and follow them without fail

4 Identify Process Handoff

5 Identify and use Collaboration Tools

Automate EVERYTHING

Do not include work arounds or exception paths

Our team can Dominate DevOps as a Distributed Team. We MUST trust each other. Our team agreement includes following the Agile Principles. We will not circumvent our documented processes.

Yes, it’s not perfect or easy, but I have what I need to be a valued part of a Distributed DevOps Team. I’m looking forward to Continuously Improving

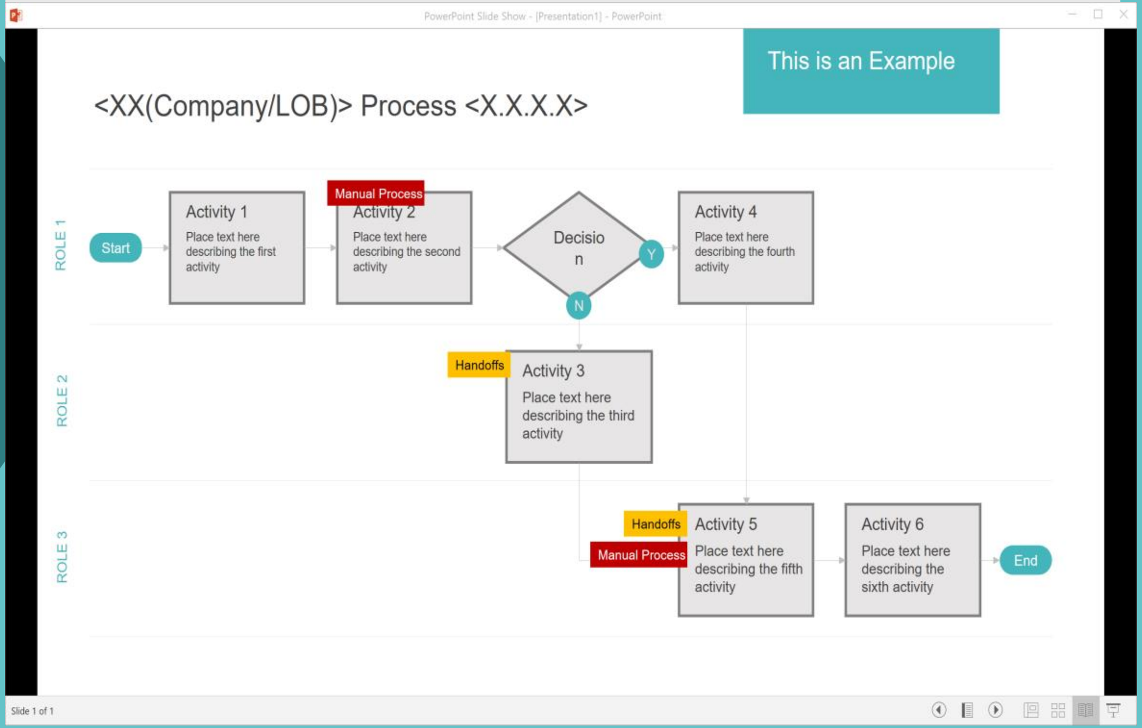
Okay, let’s review what your distributed team can do to dominate DevOps

Here are just a few things which are a part of a mature teams agreement and by consistently following help them build trust

- Respect for Culture and each other
- Automated tests are unfailingly created
- Code is checked in multiple times a day and peer reviewed
- Decisions are decentralized and are effectively made by the team
- The team has several hours that overlap ensuring the team can communicate in real time
- All team members speak and understand the same language
- The consistent use of collaboration tools

Know your team and build trust & relationships

Each team member should know where to access the documented processes which they should be adhering to over and above the team agreements



PowerPoint Slide Show - [Presentation1] - PowerPoint

This is an Example

<XX(Company/LOB)> Process <X.X.X.X>

ROLE 1

Start

Activity 1
Place text here describing the first activity

Manual Process

Activity 2
Place text here describing the second activity

Decisio n

Y

N

Activity 4
Place text here describing the fourth activity

ROLE 2

Handoffs

Activity 3
Place text here describing the third activity

ROLE 3

Handoffs

Manual Process

Activity 5
Place text here describing the fifth activity

Activity 6
Place text here describing the sixth activity

End

Slide 1 of 1

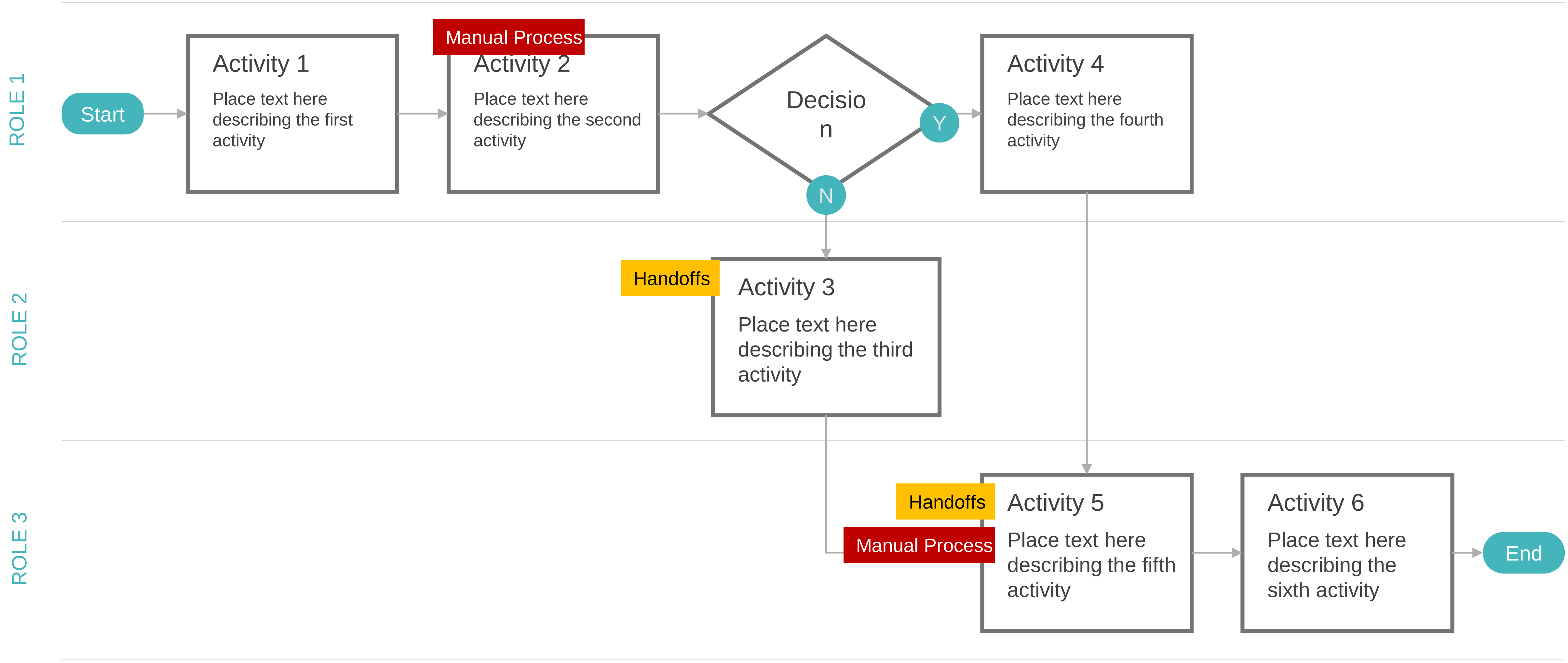
Know and understand the flow of activities in the documented process

Treasa Overton - Overton Coaching Development

<XX(Company/LOB)> Process <X.X.X.X>

This is an Example

This is what you will end up with after this exercise.



ROLE 1

ROLE 2

ROLE 3

Start

Activity 1

Place text here describing the first activity

Manual Process

Activity 2

Place text here describing the second activity

Decisio
n

Y

N

Activity 4

Place text here describing the fourth activity

Handoffs

Activity 3

Place text here describing the third activity

Handoffs

Manual Process

Activity 5

Place text here describing the fifth activity

Activity 6

Place text here describing the sixth activity

End

Treasa Overton - Overton Coaching Development

Here's what you need to get started

Know the process you've been assigned

Gather Process Information

Process maps

Knowledge content
(wiki, SharePoint docs, other)

Identify SME's to Help

Identify the SME's *(critical few)* in the processes you own and POC's from other teams you collaborate with to execute the process.

Know we are here to help!

Feel free to reach out Treasa Overton by email if you have questions @ info@agileintelligence.training

Knowing the Process Decision points allow the team to be compliant by design without the need to scramble to meet governance requirements

- Approval Roles are approving when appropriate, on time without exception
- Demonstrations are completed and features approved before moving code to production
- Risks are mitigated
- Documentation is completed and handed off
- Testing has been completed by business and features accepted
- ETC….

Know the decision points and follow them without fail

Identify your Interaction Model with your teams Enabling Partners and the handoff from your teams assigned process to a connected process

Design
Digital
Legal
HR
Communication
Change Management
Release Management
Risk
Compliance
Adaptive Planning
Vendors
Suppliers

Identify Process Handoff

Teams have so many collaboration tools available to them, the challenge is to select one that the entire team has access to and can be utilized by all, such as:

- Skype
- Zoom
- Slack
- Text
- Video Cameras and Microphones
- Agile Work Item Management Tools sucj as JIRA, Agile Central, Version One, AgileCraft
- ETC….

Identify and use Collaboration Tools

Thank you for letting me share with you my experiences on how distributed teams can DOMINATE DEVOPS!

Dominating DEVOPS with Distributed Teams

In today’s working environment of mixing office commuters and work from anywhere and to that mix throw in offshore there is very little likelihood that you will have collocated teams. How does you company implement DevOps when the Agile principle of collocated teams is pretty much extinct in most enterprise work cultures?

I propose that it isn’t as difficult as the some Agilists claim it to be, but that does mean to say it is easy either, it takes a team that has built trust and has very specific working agreements around checking in code, automating everything they can, not working on their own boxes but in a shared environment (OK checking the code they are working on into the shared repository often throughout the day)

A distributed team can (again in my humble opinion) actually provide earlier, faster delivery of quality business value through automating everything, using techniques such as test driven development or it may mean using versioning/branching to ensure that you aren’t causing conflicts with existing branches, it does however allow for the early and continuous integration of work being done so that early detection of issues can be detected. More importantly when working as a distributed team, integrating early and often lets peers review and continue coding where the last development cycle left off. What I mean by this is simple, if there are developers in India and developers in the US, they need to ensure that they are integrating often so that the major differences in time zones actually works for you instead of against you, allowing each group to peer review one another, pick up coding where the other team member left off, ensuring that faster development cycles are successful and quality increases.

Continuous Integration is one of the most important aspects of development whether the team is collocated or distributed, it is the backbone of enabling quality to be the primary benefit of using DevOps as a delivery framework. Continuous Delivery is probably the hardest part of adopting DevOps particularly if an enterprise is new to lean agile or have been transforming for quite some time and have made the leap to become more mature in agility. Continuous Delivery is made difficult because many architectural runways have not been built to enable delivery on demand, governance may take some time to catch up to the transformation and still take quite some time for products to move through the release management process, there could be a case study here, as to when do the different pieces or parts if you may, an enterprise is mature enough in its transformation to begin its adoption of the more complex aspects of DevOps, however, this is a great place and time for the distributed team to begin preparing, building, and establishing versioning and releasing into production as a toggled off product, which still enables other code to be integrated to ensure that it works seamlessly in the environment that will provide the biggest value delivery to the business, and all due to the fact that with distributed teams it gives almost double the hours of development time in a single day.

It doesn’t mean it is easy, there is compromise, understanding of each team members needs and abilities as well as some sacrifices from both side to ensure there are some overlap in work hours to collaborate.

Distributed teams can only dominate DevOps for teams that are dedicated to establishing a pattern of continuous collaboration and utilizing both free and pay collaboration tools.