



**Red Hat**  
Training and  
Certification



# DevOps Culture and Practice Enablement

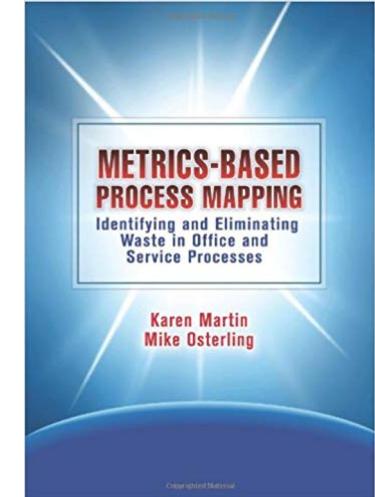
Metrics-Based Process Mapping



## The Origin

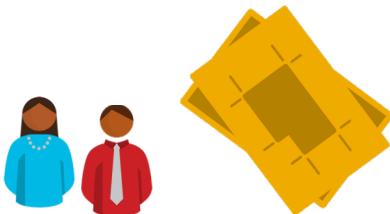
Metrics-Based Process Mapping  
by Karen Martin, Mike Osterling (2012)

[https://openpracticelibrary.com/practice/  
vsm-and-mbpm/](https://openpracticelibrary.com/practice/vsm-and-mbpm/)



Facilitator Resources: Slides & Examples

# WHICH TO USE WHEN?



## VALUE STREAM

- ▶ **Macro view** - high-level view (30k-foot)
- ▶ Future state designed by leadership
- ▶ Strategic:  
**What** are we going to do?



## METRICS-BASED

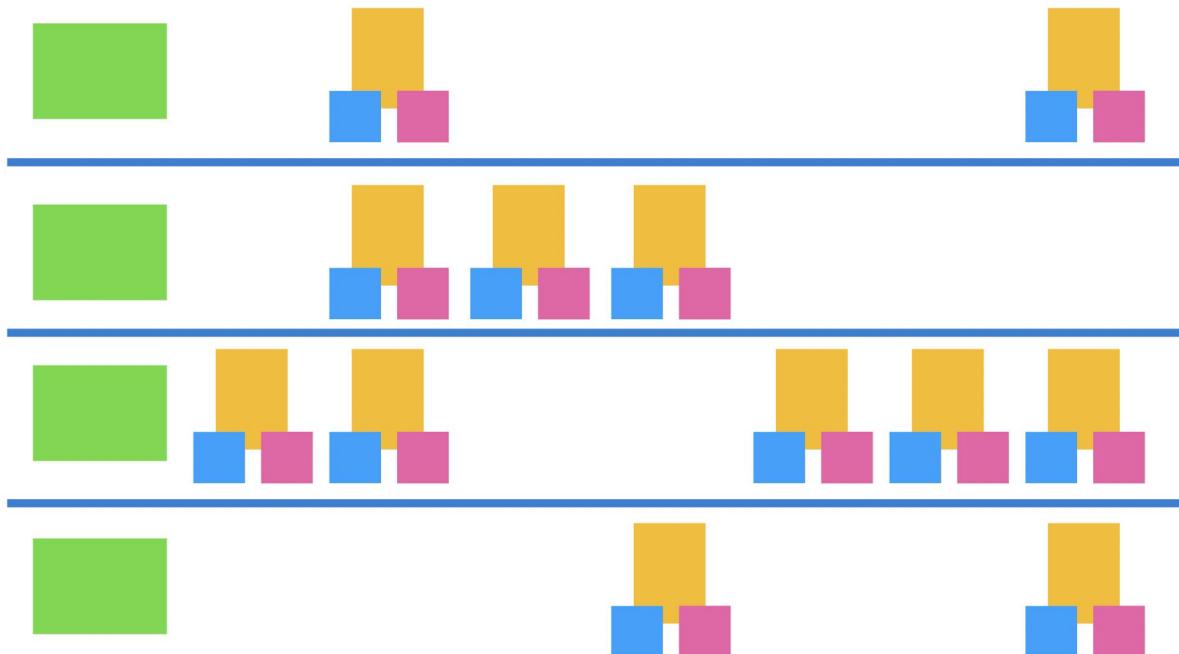
- ▶ **Micro view** - "in the weeds"
- ▶ Future state designed by frontline workers
- ▶ Tactical:  
**How** are we going to do it?

## Practice Overview

# Metrics-Based Process Mapping

4-8 HOUR | HARD | AUDIENCE MATTERS

Collect key time and quality metrics for a current process and use to identify areas for improvement. Provides a micro-level view of a process.



# Metrics-Based Process Mapping



APPROX. DURATION

4-8  
hours



DIFFICULTY

Hard



AUDIENCE

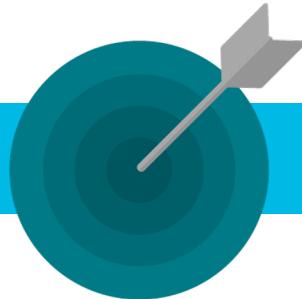
Any  
audience

## QUESTIONS

How are we spending our  
time?

Do we have bottlenecks?

How can we improve our  
processes?



## ACTIVITY GOAL

Collect key time and quality

metrics for a process.

Identify areas for improvement.



# HOW?

## THE METRICS



**PROCESS TIME** - the time it takes to actually perform the work

**LEAD TIME** - the elapsed time from when the work is made available to when it is completed and passed onto the next step

**% COMPLETE & ACCURATE** - % time downstream customer can perform the task without having to correct, to add, or to clarify the incoming work



## HOW?

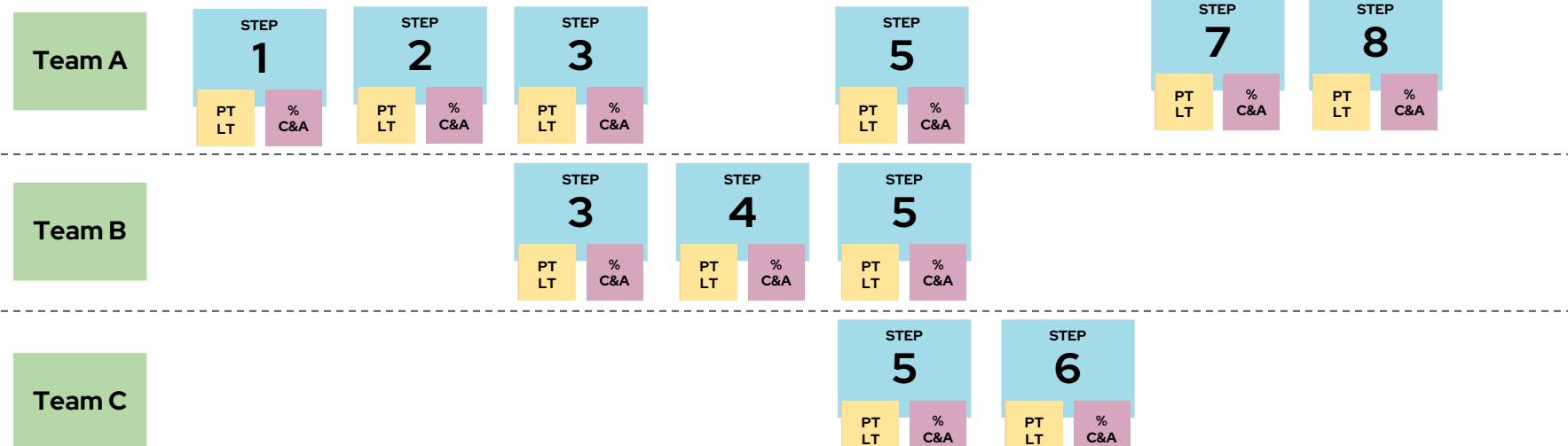
- 1. Identify fenceposts.** Clearly delineate at what point the process starts and ends.
- 2. Identify actors/workstreams.** Identify and highlight handoffs.
- 3. Ask questions** and walk through the process to list each step. Map interactions between workstreams.



# HOW?

PROCESS TRIGGER

PROCESS END





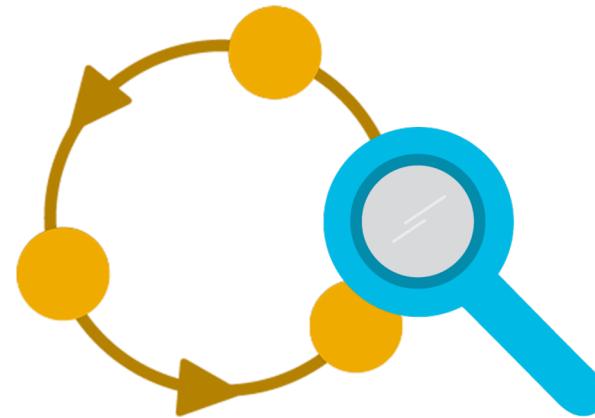
## HOW?

**Once the steps between the posts are identified:**

- ▶ Review each step and estimate lead time and process time for it.
- ▶ Ask, does this step have any impact (%) on any prior step? Note a % on the prior step.
- ▶ Accumulate the metrics to calculate the total lead time, total process time, and % complete and accurate



## OUTPUTS



**Key metrics for process time and quality**

# CURRENT vs FUTURE STATE PROCESS METRICS

## PROCESS TIME

Actual time to perform  
the work

**3.5 hrs**

**7 min**

**29x**

## CYCLE TIME

Elapsed time from Port  
Request submission to  
code deployment to DR

**50.5 hrs**

**4 hrs**

**11x**

## IMPROVEMENTS

## % COMPLETE & ACCURATE

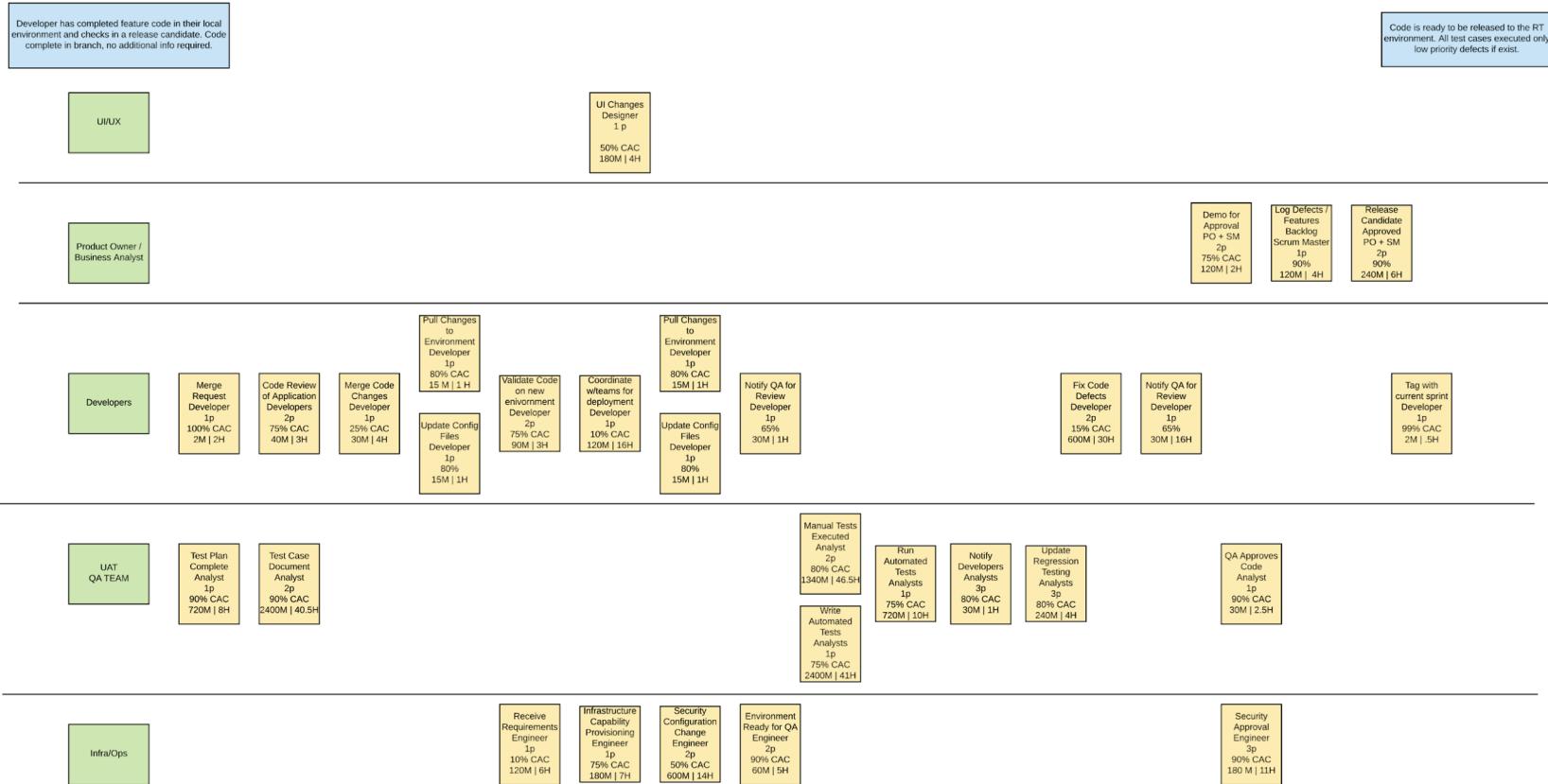
% of time a release candidate  
makes it through the entire  
pipeline without a hitch

**30%**

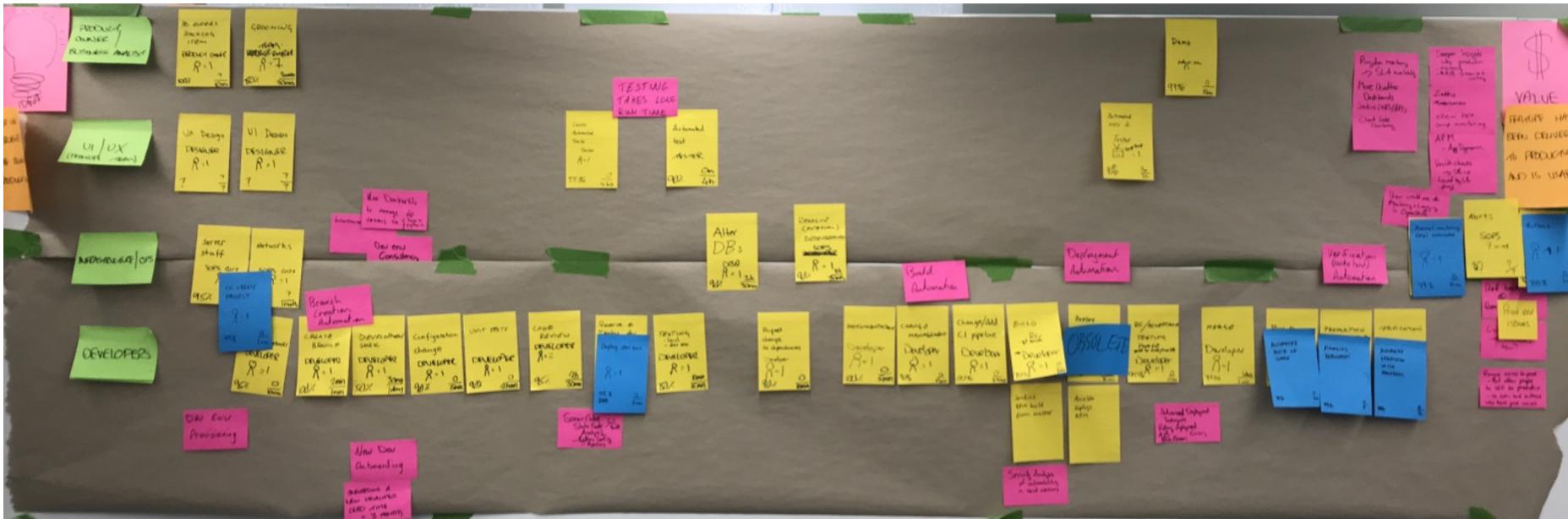
**39%**

**25%**

# SOME EXAMPLE METRICS BASED PROCESS MAPS



# SOME EXAMPLE METRICS BASED PROCESS MAPS



# FUTURE STATE PROCESS MAP

