RESULTS WASHINGTON

Agile Development and Lean Transformation, a Practical Guide

Scott Siderman, Manager
TokuSaku Consulting
October 27, 2021

得 策 TOKU SAKU CONSULTING

Agenda

- Why is Agile?
- Why is Lean?
- What is Lean Transformation?

A Few Quick Yes/No Questions

- I have been involved in a successful “Big Bang” project (System/Process change delivered all at once)?
- I have been involved with a successful Agile Development Project?
- I have been involved with a successful Lean project?
- I have been involved in a successful Lean Transformation project?

Why is Agile?

Initial Conditions

- Waterfall is prevailing
  - Phased
    - Requirements
    - Design
    - Implementation
    - Verification
    - Maintenance
  - PMI Aligned
  - Believed to be overly regulated, planned, and micromanaged

- New methods being explored
  - Rapid Application Development
  - Unified Process
  - Dynamic Systems Development Method
  - Scrum
  - Crystal Clear
  - Extreme Programming
  - Feature-driven development

Some Agile "Catalysts"

- Waterfall projects
  - Took a long time
  - Cost a lot of money
- Customers becoming more demanding
- Technology becoming pervasive

- Need for more automation
- Need for more frequent enhancement to existing functionality
- New developer tools make modular design and development easier

What is Agile? – The Manifesto

- Customer satisfaction by early and continuous delivery of valuable software
- Welcome changing requirements, even late in development
- Deliver working software frequently (weeks rather than months)
- Close, daily cooperation between businesspeople and developers
- Projects are built around motivated individuals who should be trusted
- Face-to-face conversation is the best from of communication
- Working software is the primary measure of progress
- Sustainable development, able to maintain a constant pace
- Continuous attention to technical excellence and good design
- Simplicity – the art of maximizing the amount of work not done – is essential
- Best architectures, requirements, and designs emerge from self-organizing teams
- Regularly,, the team reflects on how to become more effective, adjusts accordingly

Agile: Leaned Out Software Development

- **Customer satisfaction** by early and continuous delivery of valuable software
- Welcome changing requirements, even late in development
- **Deliver working software frequently** (weeks rather than months)
- **Close, daily cooperation between businesspeople and developers**
- Projects are built around motivated individuals who should be trusted
- **Face-to-face conversation is the best from of communication**
- Working software is the primary measure of progress
- **Sustainable development**, able to maintain a constant pace
- Continuous attention to technical excellence and good design
- Simplicity – the art of maximizing the amount of work not done – is essential
- Best architectures, requirements, and designs emerge from **self-organizing teams**
- Regularly, the **team reflects on how to become more effective**, adjusts accordingly

What is Agile? – The Reality

- No holistic requirements
  - Epics
  - User Stories
- Short development sprints
- Preponderance of management tools/software

- Anecdotal evidence – agile improves effectiveness
  - Software professionals
  - Teams
  - Organizations
- Empirical evidence - mixed and hard to find

DevOps – A Side Note

- A set of practices that combine Software Development and IT Operations
- Shares several aspects with Agile Development
- Complementary with Agile Development
- Aim is to shorten delivery time and provide continuous delivery of high-quality software

Why is Lean?

Lean is

- A method
- Uses collaborative teams
- Improves performance
- Systematically eliminates waste
- Often linked with Six Sigma (focuses on reducing variation)

The Eight Wastes

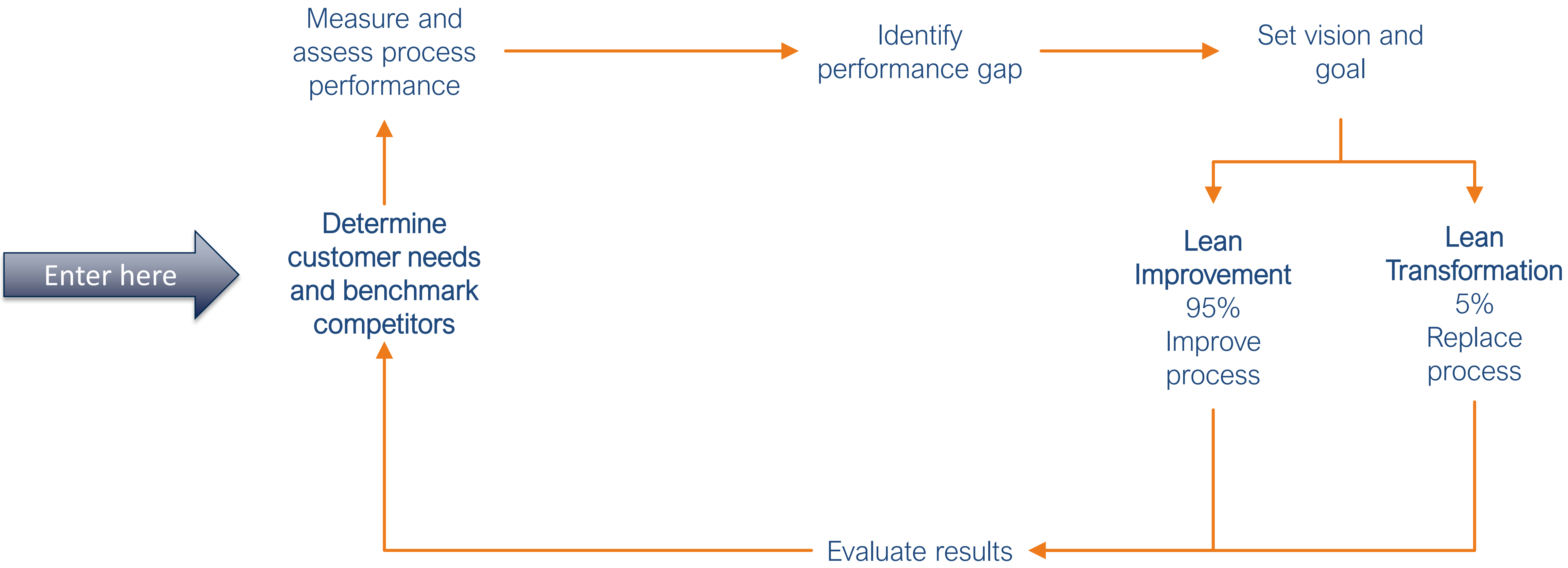
- Defects
- Overproduction
- Waiting
- Non-utilized Talent
- Transportation
- Inventory
- Motion
- Extra-processing

Practical Lean Concepts

- Focus on customer satisfaction and the delivery of value
- Often focused on frequent, incremental improvements
- Close, daily improvement discussions between managers and workers
- Projects are built around motivated individuals who should be trusted
- Face-to-face conversation is the best from of communication

- Projects are focused on specific, more difficult to improve, issues
- Workshops are a typical solution environment
- Continuous attention to performance excellence and effective metrics
- Regularly, the team reflects on how to become more effective, adjusts accordingly

The Lean Virtuous Cycle



Enter here

Determine customer needs and benchmark competitors

Measure and assess process performance

Identify performance gap

Set vision and goal

Lean Improvement
95%
Improve process

Lean Transformation
5%
Replace process

Evaluate results

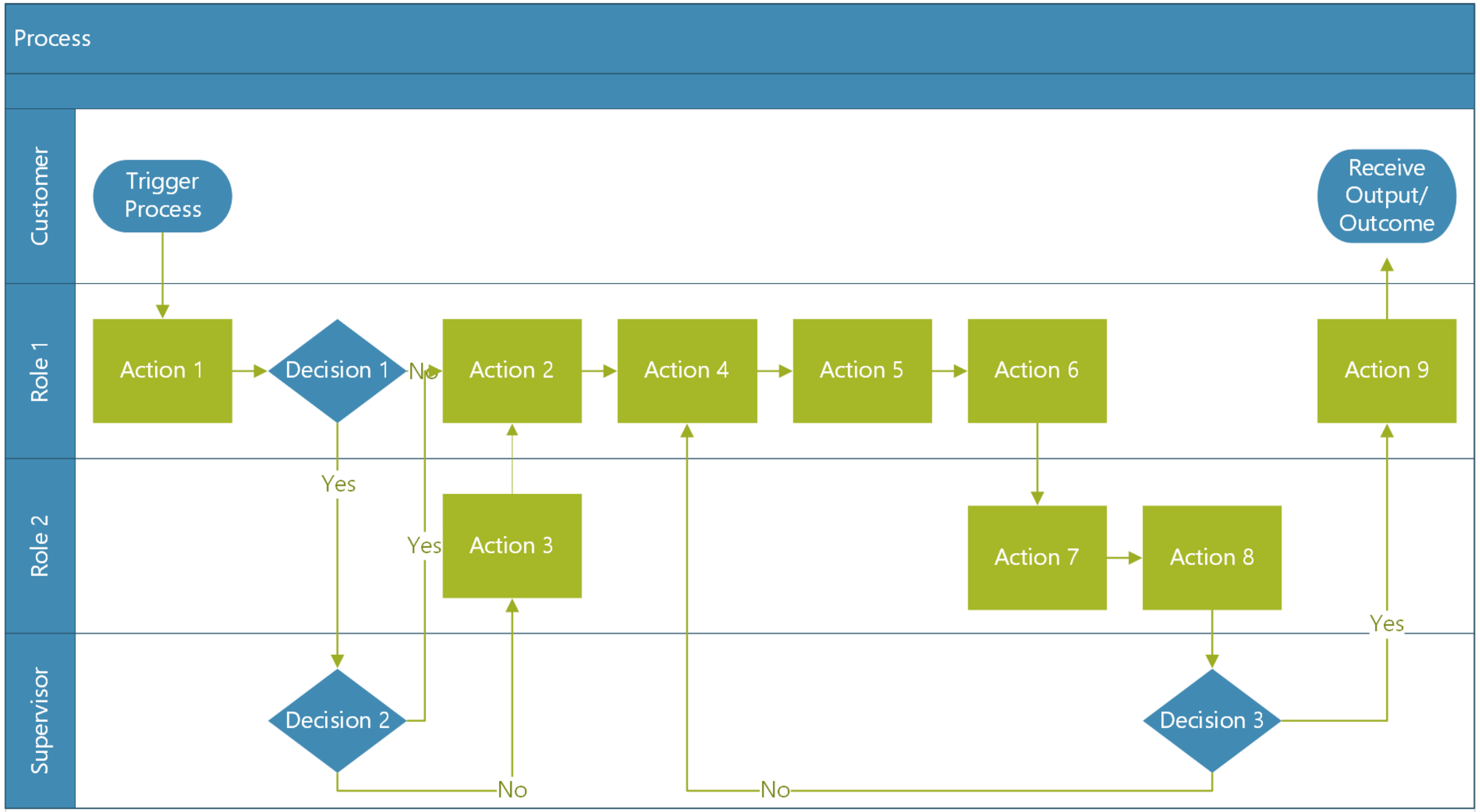
Adapted rom “Reengineering: the Implementation Perspective”, Hammer and Company, 1995

What is Lean Transformation?

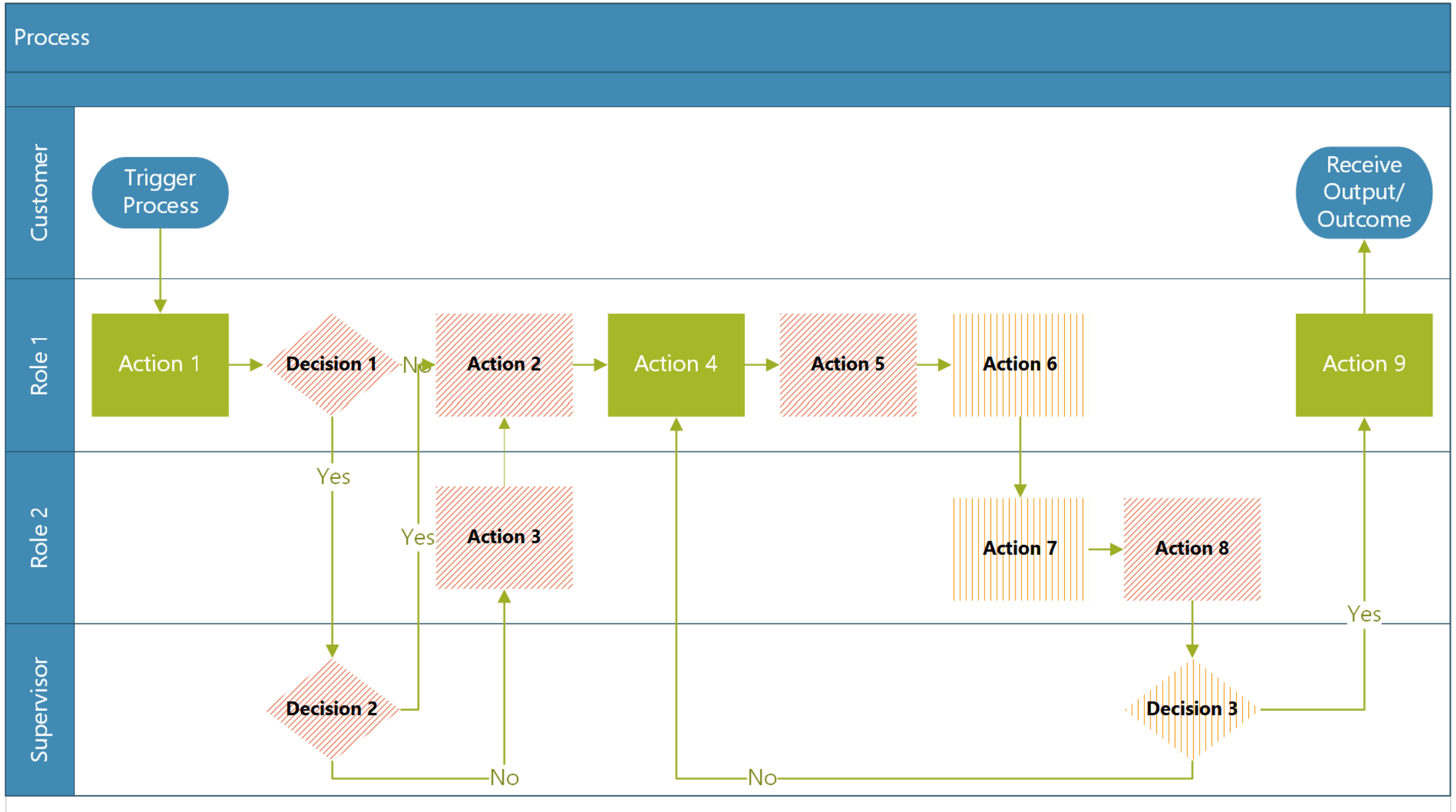
Transforming Processes Using Lean

- Document current process
- Classify each action
  - Value Added
  - Business Value Added
  - Waste
- Organize Value Added actions
- Realign and add back selected Business Value Added actions
- Attempt to eliminate waste entirely

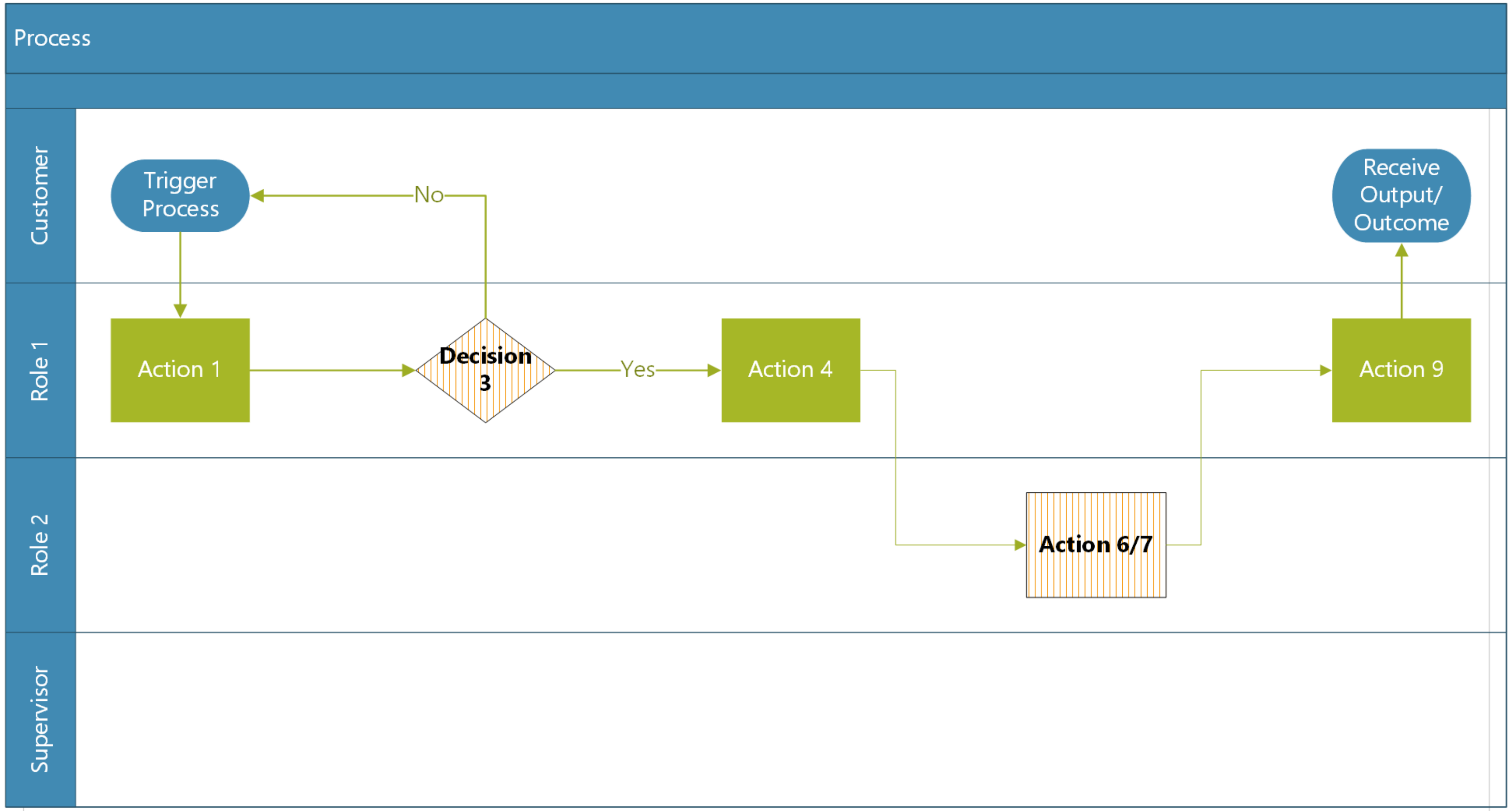
Current State



Value Added Analysis



Future State (Target)



Traditional Migration Approach

- Steps
  - Sketch solution design
  - Develop complete specifications
  - Conduct ROI analysis
  - Formulate implementation plan
  - Obtain resources
  - Construct components
  - Integrate components
  - Train participants
  - Install and cut over

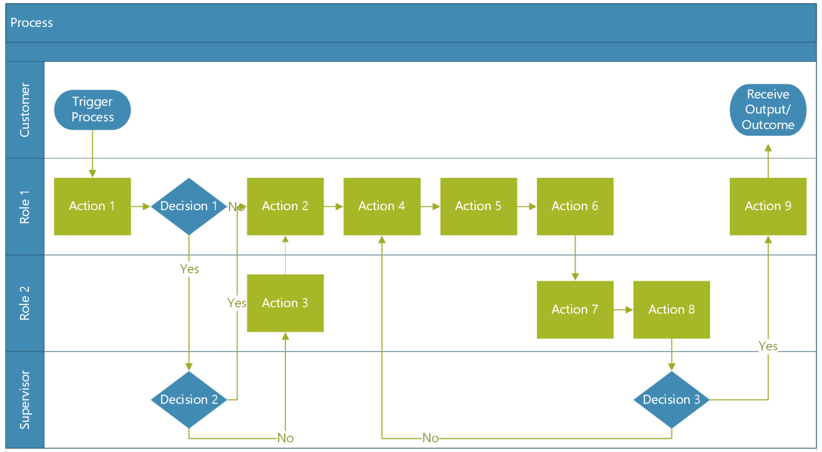
- Characteristics
  - Replete with handoffs
  - Fragmented
  - Monolithic
- Performance
  - Slow
  - Unpredictable
  - Inflexible

A Release is

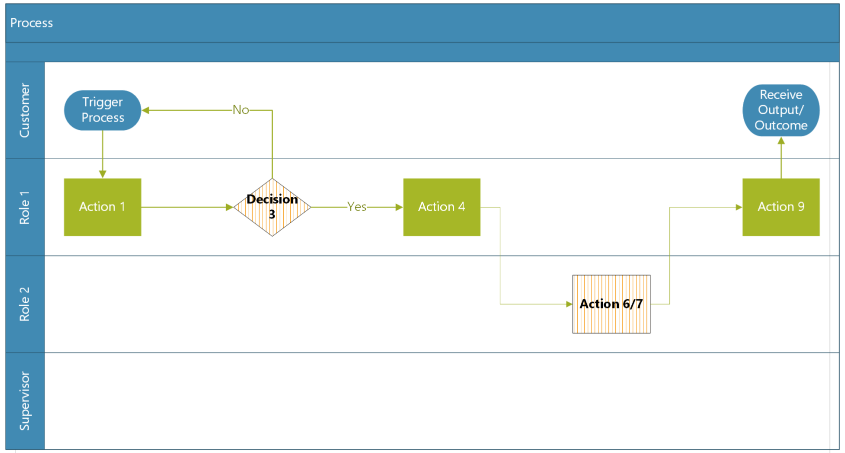
- The increment of implementation
- A subset of the end state
- Typically implemented in less than 9 months
- Holistic
- A chunk of business value

Creating a Release – Identify Improvements

- How will we get from our current to future state?
- Identify all the changes needed to realize the new design
- Remember that the future state is a (moving) target



Current

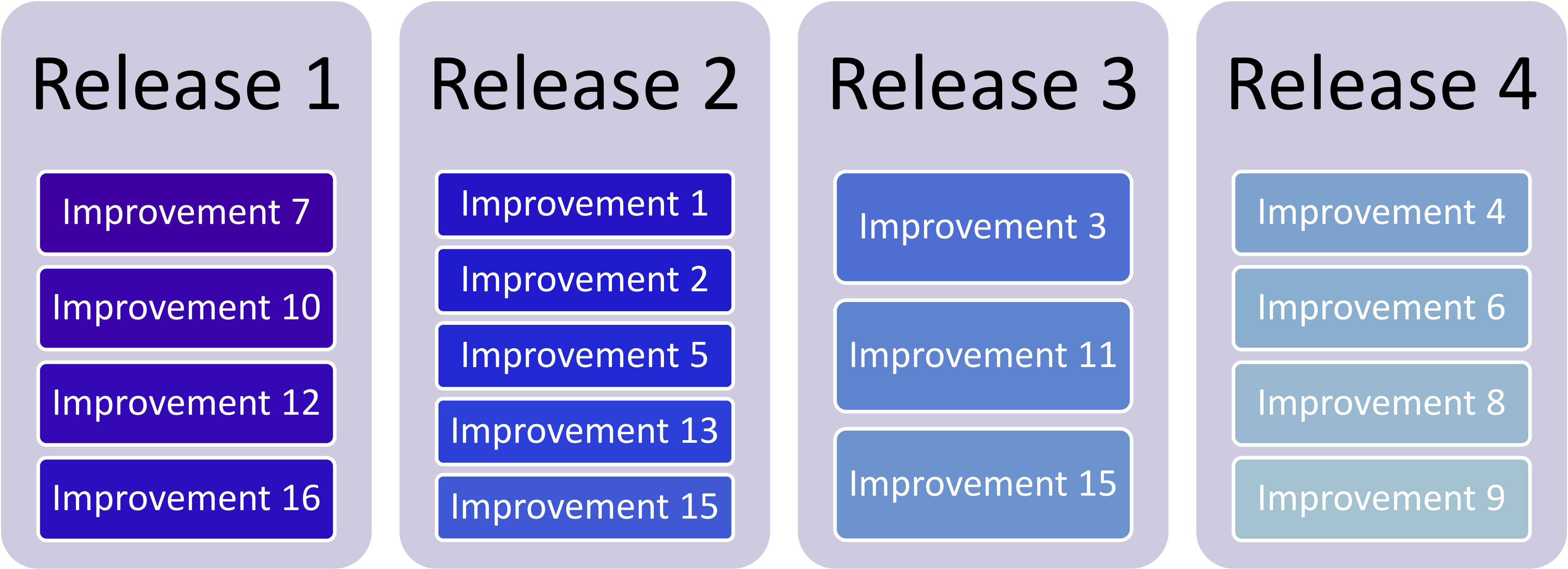


Future

Creating a Release – Identify Improvements

- Lean efforts end with a cataloging of improvements
- Then, the differences show up
  - Improvement efforts create a time-ordered list
  - Transformations chunk all improvements into releases
- Side note: Quick Hits
  - Easy and fast to implement
  - Provide early value and wins
  - Consistent with the future state
  - Implemented while developing and prior to the start of the formal plans

Creating Releases – Affinitize Improvements



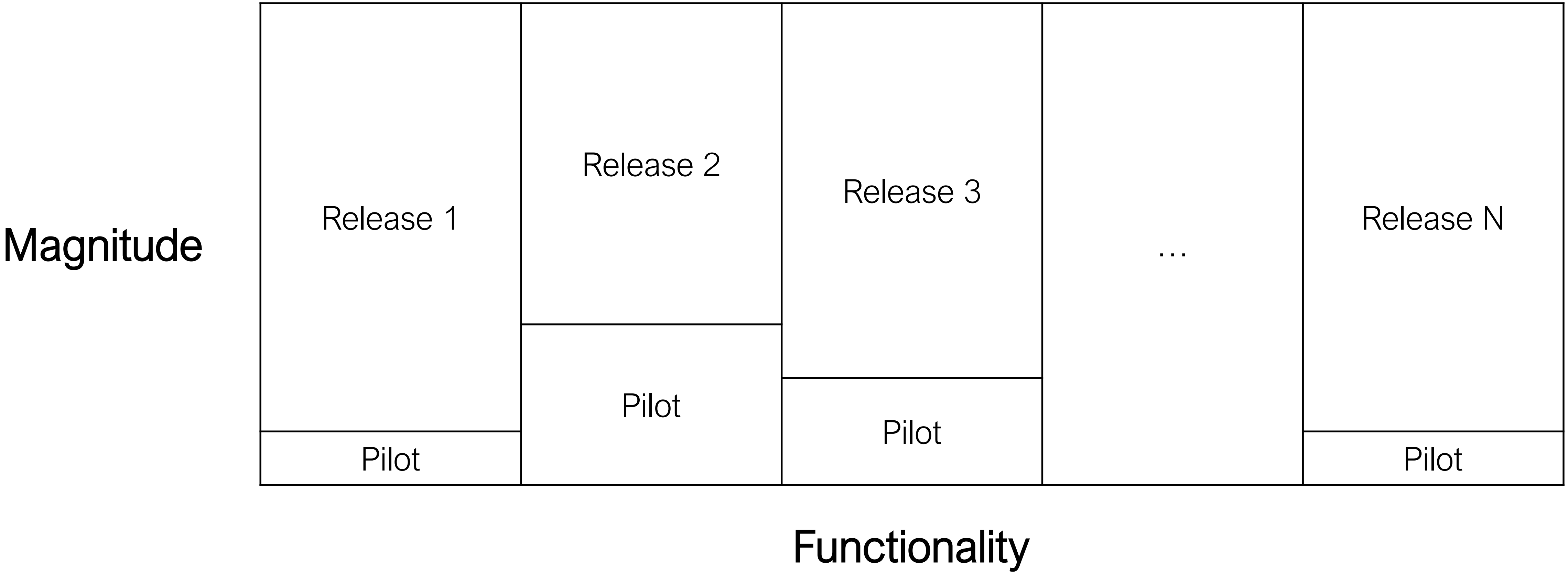
Release 1
- Improvement 7
- Improvement 10
- Improvement 12
- Improvement 16

Release 2
- Improvement 1
- Improvement 2
- Improvement 5
- Improvement 13
- Improvement 15

Release 3
- Improvement 3
- Improvement 11
- Improvement 15

Release 4
- Improvement 4
- Improvement 6
- Improvement 8
- Improvement 9

Migrating from Current to Future States: Release-based Implementation



Magnitude

Release 1

Pilot

Release 2

Pilot

Release 3

Pilot

…

Release N

Pilot

Functionality

Magnitude “Dimensions”

- Geography
- Product line
- Market segment
- Customer segment
- Distribution channel
- Business unit
- Employee level

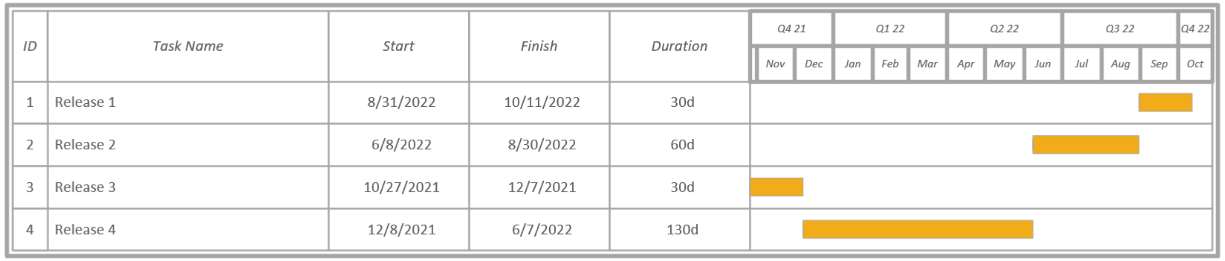
Why Releases?

- Speed
  - Deliver meaningful change more quickly
  - Produce results to “pay for” subsequent releases
- Learn in real time
  - Results of early releases advise subsequent releases
  - React to outside changes as needed
- Integrated, workable solutions
  - Demonstrate progress
  - Ease resistance to change
- Parallelism
  - Avoid sequential implementation of a long list of improvements
  - Requires good intrateam communication
- Pay it forward

Implementation Planning Options

Typical

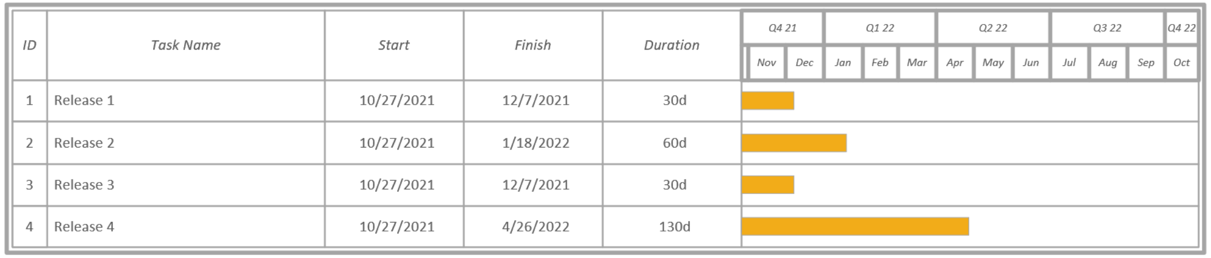
- Prioritized
- Fewest resources throughout
- Slowest to complete



ID	Task Name	Start	Finish	Duration
1	Release 1	8/31/2022	10/11/2022	30d
2	Release 2	6/8/2022	8/30/2022	60d
3	Release 3	10/27/2021	12/7/2021	30d
4	Release 4	12/8/2021	6/7/2022	130d

Race

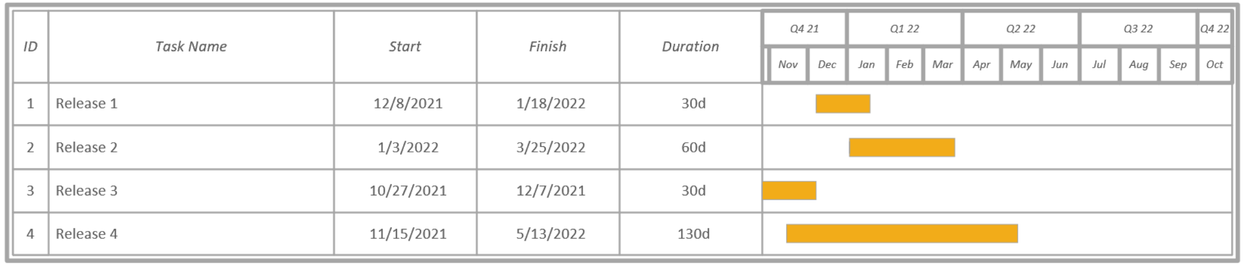
- Start simultaneously
- Tapering resources
- Fastest to complete



ID	Task Name	Start	Finish	Duration
1	Release 1	10/27/2021	12/7/2021	30d
2	Release 2	10/27/2021	1/18/2022	60d
3	Release 3	10/27/2021	12/7/2021	30d
4	Release 4	10/27/2021	4/26/2022	130d

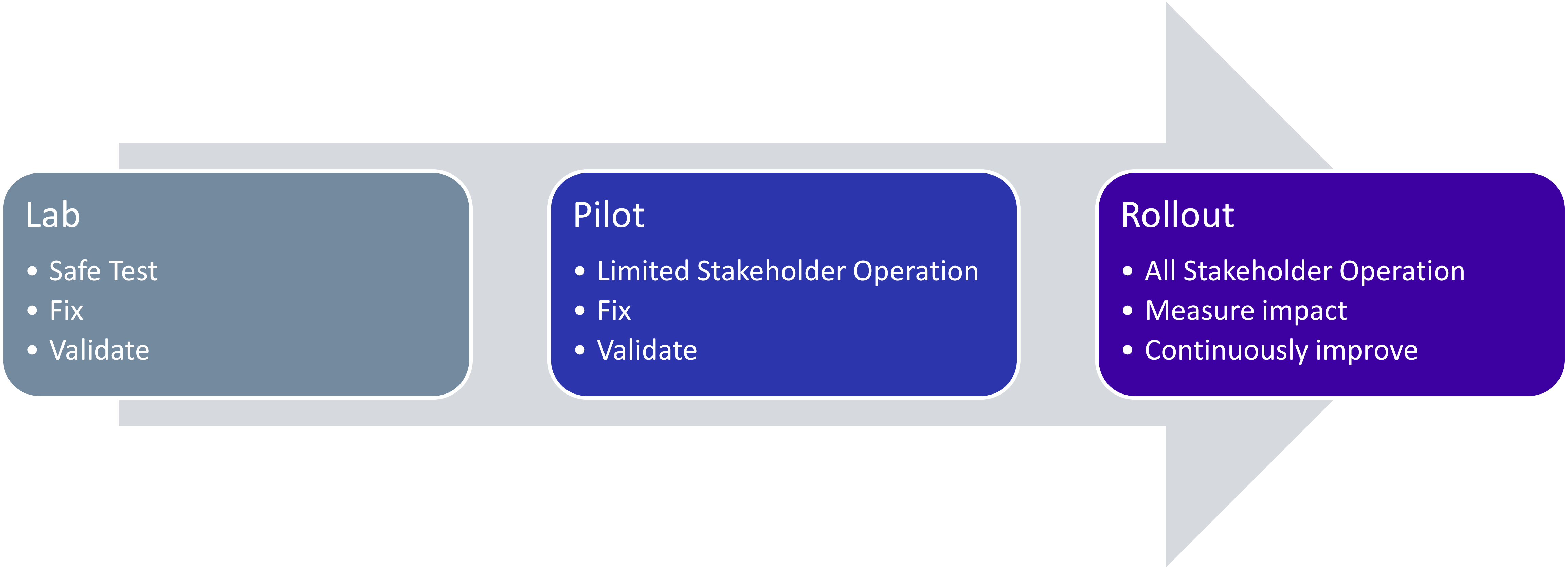
Semi-parallel

- Ordered by delivery of value
- Balancing of resources
- Faster



ID	Task Name	Start	Finish	Duration
1	Release 1	12/8/2021	1/18/2022	30d
2	Release 2	1/3/2022	3/25/2022	60d
3	Release 3	10/27/2021	12/7/2021	30d
4	Release 4	11/15/2021	5/13/2022	130d

Inside a Release



Lab

- Safe Test
- Fix
- Validate

Pilot

- Limited Stakeholder Operation
- Fix
- Validate

Rollout

- All Stakeholder Operation
- Measure impact
- Continuously improve

The Lab – “Hidden” Part of the Release

- A Lab precedes all pilots
- In the lab we
  - Create a simulation of real life
  - Experiment with detailed design
    - Process
    - Enabling Automation
  - Share progress with stakeholders in real time
  - Iterate until “done”

- Benefits include
  - Safety
  - Insulation for failure
  - Low-risk learning
  - Not prematurely making large-scale change
    - Cost
    - Exposure
    - Need and/or inability to roll back

Questions?

Scott Siderman, Manager
(206) 364-0296
scottsiderman@tokusaku.cc

Review

Characteristic	Traditional	Agile	Lean	Lean Transformation
Overall Scope	Automation	Automation	Process	Process and Enabling Technology
Implementation Speed	Resource dependent	Fast	Fast	Fast
Focus	System	Customer	Customer	Customer
Payoff	Back-end loaded	Frequent(?)	Frequent	Quick
Amount of Value	Large(?)	Unclear	Small	Large
Completion	On Schedule	As available	Continuous	Periodic
Structure	Sequential	Parallel	Sequential	Parallel
Goal	Perfection	Features	Continuous Improvement	Holistic Improvement

得策 TOKU SAKU

CONSULTING

Scott Siderman, Manager
(206) 364-0296
scottsiderman@tokusaku.com