

1. We increase flow by making work visible, by reducing batch sizes, and intervals of work and by building quality in, preventing defects from being passed to downstream work centers.
2. The goal is to decrease the amount of time required for changes to be deployed in production and to increase the reliability and quality of those services.

3. By putting all work for each work center in queues and making it visible - all stakeholders can more easily prioritize work in the context of global goals.
4. Controlling queue size (Limiting Work in Progress) is an extremely powerful management tool as it is one of the few indicators of Lead time. Limiting WIP also makes it easier to see problems that prevents the completion of work.
5. We achieve a fast and smooth flow by performing work in small batch sizes. Large batch sizes result in high levels of WIP and high levels of variability and flow that cascade through the entire process – resulting in long lead times and poor quality.
6. One of the factors in longer lead times is the large number of handoffs which we often see in a value stream. We must strive to reduce the number of handoffs by automating significant portions of work or by reorganizing teams that can deliver value to the customers themselves.
7. Eliminate waste in software development
 - a. Partially done work that become obsolete and lose value as time progresses
 - b. Extra processes that add effort and increase lead times
 - c. Extra features that add complexity and effort to testing and managing functionality
 - d. Task switching which leads to additional time and effort
 - e. Waiting which increases cycle time and prevents the customer from getting value
 - f. Motion, such as handoffs, creates motion wastes and often requires additional communication to resolve ambiguities
 - g. Fast resolution of defects because the longer the time between defect creation and defect detection, the more difficult it is to resolve the defect
 - h. Nonstandard or manual that work cause issues
 - i. Heroics sap the energy and enthusiasm from the team

Feedback

1. We make our system of work safer by creating fast, frequent, high quality information flow throughout the value stream and our organization which includes feedback and feedforward loops.
2. Complex systems typically have a high degree of interconnectedness of tightly coupled components and system level behavior – and failure is inherent and inevitable in such complex systems – hence the need for designing a safe system of work.
3. It is not sufficient to merely detect issues when the unexpected happens – we must also swarm them, mobilizing whoever is required to solve the problem. Swarming is required to prevent the problem from going downstream, and preventing the work center from starting new work.
4. To enable fast feedback in the technology value stream, we must create the equivalent of an Andon cord and the related swarming response.
5. It is impossible for a developer to learn anything when someone yells at them for something they broke six months ago- that is why we need to provide feedback to everyone as quickly as possible, in minutes, not months.

Continual Learning and Experimentation

1. High performing manufacturing operations promote learning – the work is not very rigidly defined, the system of work is dynamic, they conduct experiments to generate new improvements.
2. Three types of culture
 - a. Pathological - large amounts of fear and threat. People often hoard information, withhold it, distort it for their own good. Failure is hidden.
 - b. Bureaucratic - rules and processes, often help individual departments to hold onto their turf. Failure is processed through a system of judgement resulting in either punishment or justice and mercy.
 - c. Generative - actively seeking and sharing of information to better enable the organization to achieve its mission. Responsibilities are shared and failure results in reflection and genuine inquiry.
3. We improve daily work by explicitly reserving time to pay down technical debt, fix defects, and refactor and improve problematic ideas of our code and environments.
4. When new learnings are discovered locally, there must also be a mechanism to enable the rest of the organization to use and benefit from that knowledge – convert tacit knowledge into explicit, codified knowledge which becomes someone else's expertise through practice.
5. Lower performing organizations buffer themselves from disruptions in many ways – they bulk up or add flab (an increased inventory buffer, hiring more people than required – often leading to increased costs). High performing organizations achieve the same results by improving daily operations, introducing tension to elevate performance, and engineering more resilience into their system.
6. Leadership helps coach the people conducting experiments with questions such as –
 - a. What was the last step, what happened?
 - b. What did you learn?
 - c. What is your next target condition?
 - d. What is your next step?
 - e. What obstacle are you working on now?
 - f. What is the expected outcome?