

# Value Stream Mapping

Learn how this analysis technique can optimize your CD pipeline.



BY JUNI MUKHERJEE

## Browse topics

### Continuous Delivery Principles

- Overview
- Continuous integration vs. continuous delivery vs. continuous deployment
- Business Value of Continuous Delivery**
- Value Stream Mapping**
- Configuration management: definition and benefits
- DevSecOps: Injecting Security into CD Pipelines
- Feature Branching Workflows for Continuous Delivery

### Branching Workflows for Continuous Delivery

### Super-Powered Continuous Delivery with Git

### Why agile isn't agile without continuous delivery

### What is cloud computing? An overview of the cloud

### How infrastructure as code (IaC) manages complex infrastructure

### Cloud Bursting

### Feature Flags

### Platforms as a Service

### Continuous Delivery Pipeline 101

### What is Continuous Integration?

### Software testing for continuous delivery

### What Is Continuous Deployment?

### Microservices and Microservice Architect

### Bitbucket CI/CD tutorials

### Continuous Delivery articles

## What is value stream mapping?

Value stream mapping (sometimes called VSM) is a lean manufacturing technique to analyze, design, and manage the flow of materials and information required to bring a product to a customer. Also known as "material and information-flow mapping", it uses a system of standard symbols to depict various work streams and information flows. Items are mapped as adding value or not adding value from the customer's standpoint, with the purpose of rooting out items that don't add value.

Value stream mapping can be used to improve any process where there are repeatable steps – and especially when there are multiple handoffs. In manufacturing, handoffs are easier to visualize because they usually involve the handoff of a tangible deliverable through stations. If, for example, a problem arises when assembling a vehicle, line workers can see the physical parts accumulating and jamming up a certain part of the assembly line. They can then stop the line to solve that problem and get the process flowing again.

The application of value stream mapping – also referred to as "visualizing" or "mapping" a process – isn't limited to the assembly line. Lean value stream mapping has gained momentum in knowledge work because it results in better team communication and more effective collaboration.

Much of the waste in knowledge work occurs in the handoffs (or wait time) between team members, not within the steps themselves. Inefficient handoffs lead to low productivity and poor quality. Value stream mapping helps identify waste and streamline the production process. Value stream mapping can be applied to both the product and customer delivery flows. Product flow focuses on steps required to optimize product delivery and completion. The customer flow focuses on the steps required to deliver on end user requests and expectations.

If you're familiar with [continuous delivery](#), then you likely already have an idea of how value stream mapping can apply to – and improve – that process. But before we dive into that topic, let's take a look at some of the pros and cons of adopting value stream mapping.

## The history of value stream mapping

The origins of value stream mapping are often attributed to Toyota Motor Corporation. However, this is a murky topic. Toyota may have adopted it from other origin sources or it may have grown organically from shared ideas in the lean manufacturing community. Early versions of diagrams revealing the flow of materials and information can be found as early as 1918 in a book called *Installing Efficiency Methods*, by Charles E. Knoepfle.

Inside Toyota the practice was called "material and information flow mapping" and was done almost as an afterthought. Toyota's success and use of lean manufacturing practices helped promote value stream mapping as a modern best practice for high efficiency business teams during the 1990s.

## The benefits of value stream mapping

Value stream mapping is critical for business sustainability. Here's why:

### RELATED TUTORIAL

[Continuous Delivery Tutorial](#)

[Try this tutorial →](#)

### SUBSCRIBE

Sign up for more articles

Email

[Subscribe](#)

- Reducing or eliminating waste can improve your company's bottom line. As a bonus, you discover the root cause and the source of the waste.
- Once wasteful handoffs are identified as part of value stream visualizers, your teams can consciously improve behavior, culture, communication, and collaboration.
- Teams discard individual opinions and prioritize based on the customer's perspective.

## The challenges of value stream mapping

Value stream mapping can be wasteful in itself, if you are not careful. Here's how you can avoid common pitfalls:

- The LOE (level of effort) to conduct value stream mapping should be balanced with the potential value and savings. Essentially, keep an eye on the return on investment ROI(return on investment) from the start.
- Involve experienced people from the business side and product side in conducting value stream mapping since the mapping process could be vastly cross-functional and complex.
- Fear and uncertainty are common symptoms when value stream mapping is conducted, and so the process of identifying waste can be intense.
- Improving a step here and a step there will rake in savings for sure. However, it may not directly translate to a bottom line improvement until a full walkthrough is completed. Having said that, baby steps are often a great way to start.
- Don't rush to use professional charts, tools, and symbols right away. First, sketch with a pencil or use a whiteboard to outline the idea. Once the dust settles, formalize the map appropriately. Remember, you are trying to cut waste and not create any more than you already have.

Overall, doing value stream mapping is fine, yet overdoing it can be problematic.

## Value stream mapping use cases

Let's briefly look at how value stream mapping can bring value to various industries. The domain determines the process items that flow through the value stream map.

In a supply chain, value stream mapping can root out costly delays leading to a finished product. In manufacturing, value stream mapping helps identify waste by analyzing each step of material handling and information flow. The process items that flow through the value stream are materials.

In service industries, value stream mapping facilitates effective and timely services for external customers, whereas inside administration and offices, it facilitates services for internal customers. In healthcare, value stream mapping ensures that patients are effectively treated with high-quality care. The process items that flow through the value stream are customer needs.

## How value stream mapping identifies and reduces waste

Value stream mapping originated in the enterprise manufacturing industry. As an example, let's imagine an automobile factory receives orders for new cars and needs raw materials to produce them. The company uses value stream mapping to outline the steps required to produce a new car.

After reviewing car production steps, the company identifies a handoff stage in the development that appears wasteful. This handoff stage requires a forklift to move raw materials from one side of a warehouse to the production line. However, this move has safety risks and is time consuming. From this insight the company decides to permanently move the raw material storage directly adjacent to the production line. This increases efficiency and potentially removes the requirement of the forklift altogether!

Lean manufacturing has a set of seven types of waste generation

### **Overproduction**

Overproduction is a catalyst to many other forms of waste. When a manufactured product is overproduced it leads to further waste through unnecessary costs like extra storage, wasted raw materials, and capital frozen in useless inventory.

### **Inventory**

Inventory waste is the liability cost that comes with storing and preserving a surplus inventory. This waste includes waste of space for housing inventory, waste of rent for storage space, waste of transportation costs, waste caused from deterioration of housed products.

### **Motion**

Motion waste is the cost of all the motion by person or machine that could be minimized. The previous example we demonstrated with the forklift and supply location is a great example of motion waste and optimization. Motion waste has many wasteful byproducts, including pollution, fuel waste from operating vehicles, maintenance repairs, and costs from equipment breaking down.

### **Defects**

Accidents do happen, and they can be expensive. Defect waste management is the effort to identify and mitigate accidents and imperfections that lead to defective final products. Defects are costly as they need to be replaced, may have additional recycling costs, or may be a total loss of raw materials.

### **Over-processing**

Over-processing waste refers to any step of the manufacturing component that can be deemed unnecessary. Examples include adding features users did not ask for Or polishing areas of a product that may not be visible to a user.

### **Waiting**

Waiting waste is the cost of any step in the manufacturing processing that is slow and causes a delayed reaction to the final output. Waiting causes expenses in lighting, heating, cooling, and the risk of materials, or contracts expiring.

### **Transport**

Transport waste is very similar to motion waste. Transport waste deals with external transport movement between multiple locations or third-party partnerships where motion deals with internal movement in the same location.

A software development organization doesn't deal with physically moving raw materials around warehouses to build a finished product. Software development entails shipping ideas into tangible user experiences that provide value to the customer.

Value stream mapping for a software enterprise looks at the flow of taking "idea input" from sources like customer support, sales requirements, competitor analysis and delivering that as valuable output to the end customer. The software development value stream mapping flow stages are primarily concerned with cross-team communication.

The user requests a feature, product teams design functionality, engineers receive the design and build the software, and the software is shipped to the end user. Value stream management for software can be used to identify points of waste between these stages.

The following is a list of seven types of waste found in software development or other creative work.

### **Partially completed work**

This occurs when software is pushed out in an incomplete state. It may happen due to a lack of complete specification, or lack of automated test coverage. Partially completed work also causes a cascade of other waste since additional

work also causes a cascade of other waste since additional work is needed to push more updates and fill in the missing functionality.

### Extra features

Often referred to as “feature creep”, extra features cause waste by doing more work than is required. Extra features are features directly not requested by users but cooked up internally on a hunch or speculation. Extra features may present themselves as well intentioned but often are a byproduct of a disconnect from actual customer needs.

### Relearning

Relearning waste can also occur from lack of internal documentation. If a software failure or outage occurs it is a best practice to investigate and document why the outage happened and how it was remedied. If a failure occurs again and it has not been documented, there will need to be another investigation and remediation.

Relearning waste also occurs when a team or individual needs to overcome the learning curve of an unfamiliar technology. Tech trends rapidly come and go. Flavor of the month frameworks and libraries get jumped on by junior developers pumped by market trends and hype. Even though an organization already knows how to build a certain feature they may have to relearn how to build it in new framework.

### Handoffs

Handoff waste occurs when project owners change when roles change or there is employee turnover. Key team members leave and a project gets handed to a team member without context. This scenario is hard to avoid. Handoff otherwise occurs from poor management and changing team member priorities while in action.

Handoff waste can also occur through communication pipelines. For example, in a [DevOps team](#), the development team can integrate more closely with the operations team to help prevent any communication errors when passing off a product to be maintained. This is an example of avoiding handoff waste.

### Delays

Delays usually occur when there are tightly coupled dependencies on a project. Downstream execution on a project may be halted due to a dependency on an upstream decision or resource. While it's best to avoid dependencies between these tasks, it can be challenging to perfectly decouple tasks. A delay in one task may cost delays in dependants. Delays can cause a cascade of waste. Software development often happens at a rapid pace and tasks are distributed amongst team members.

### Task switching

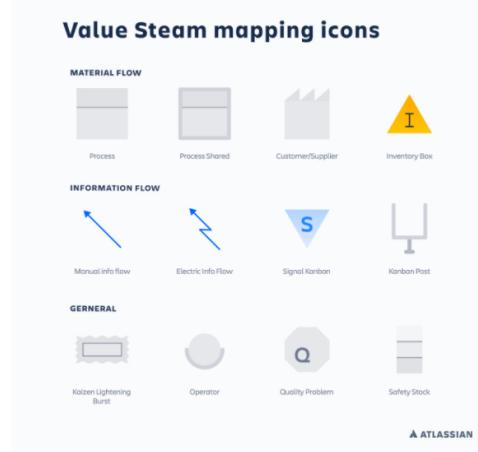
Task switching waste has similar qualities as handoff waste. Where handoff waste occurs when tasks switch owners between team members, task switching waste happens to an individual. Mental context switching is expensive. There is a mental cadence or “flow” that software engineers achieve to optimally produce good code. Efficient organizations work to optimize this mental state for their engineers. Inefficient organizations bombard their engineers with non-critical distractions like meetings and emails that disrupt their workflow.

### Defects

Defect waste happens when bugs are pushed in software. Defects are similar to partially completed work but can be more wasteful because defects are unknown and partially completed work is usually known ahead of time. Defects may be identified by customers and then reported to customer support, which can be an expensive pipeline that causes delays and task switching.

## The value stream mapping symbols

There are standard symbols for outlining value stream mapping.



## How to create a value stream map - one step at a time

### 1. Decide the problem you are solving for

What problem are you solving from the customer's standpoint? Do your customers feel like it takes too long to deliver new features or improvements to a product? Publish the problem statement and get everyone on the same page.

### 2. Empower the right team

Empower a mature and experienced team who can skillfully address these problems in a timely fashion. The C-suite should set aside enough budget to ensure that execution is uninterrupted.

### 3. Bound the process

Once the problem statement is published, limit the scope of your value stream mapping accordingly. You may not need to map the release process in its entirety, and focus on a particular area instead.

### 4. Map the bounded process

Be sure to review the bounded process. This can make a difference, since firsthand experience cannot be substituted by (possibly biased) narratives and (possibly incomplete and inaccurate) documentation done by others.

Define the steps. I conduct a value stream mapping analysis multiple times. While this may sound redundant, I have found missing pieces in the second pass that were not exposed in the first pass. And when we investigated further, skeletons fell out of the closet during the third (and final) pass!

### 5. Collect process data

As you conduct value stream mapping, note the process data in the data boxes of the map. Process data includes (but is not limited to) the number of people involved, the average number of working hours, cycle time, wait time, uptime, downtime, and more.

### 6. Create a timeline

Map out process times and lead times.

### 7. Assess your current map

Be inquisitive. Do your teams have multiple dependencies with each other? Is your lead time too long? If it is, is it because your test suites don't (or can't) run in parallel? Do you have stable environments, or do you observe intermittent test failures that the teams cannot reproduce?

Or, you may have process steps that you think are valuable but don't mean anything to the customer. Regarding the information flow, look for stagnation and drag in the flow. Note whether it was a push versus a pull.

### 8. Design the future map

You may not be able to nail a full and final version, and that's okay. Make sure your new map aligns with the

that's okay. Make sure your new map aligns with the company's vision.

Also, nothing is set in stone. Based on customer needs, make continuous adjustments.

### 9. Implement the future map

Follow the value stream mapping of the future and validate that it makes better sense for the customers. It should have solved the problem statement that you started with. Monitor KPIs regularly and learn from trends. Make sure everyone is rowing in the direction of customers.

If you're interested to see what the finished product looks like, here's a [value stream map example](#).

## The application of value stream mapping to continuous delivery

In software development, value stream management can reveal inefficiencies from idea to production, including feedback loops and rework. It can help reduce the number of steps and the need for rework. Mapping your process can help you visualize where handoffs occur so you can also discover where wait time keeps work from moving through your system.

By definition, continuous delivery (CD) doesn't need to make use of value stream mapping and it is perfectly possible to design and [implement a CD pipeline](#) without knowledge of value stream mapping.

With proper implementation, value stream mapping fosters a culture of continuous improvement that has been proven effective in software engineering and operations. The map illustrates the outcomes of the value stream analysis, providing a visual tool to facilitate understanding and communication.

## Why bring value stream mapping to your team?

Value stream mapping can be applied to industries that are looking to improve their processes across all business functions. Visualizing handoffs help optimize the flow and help generate savings. Without the visualization, expect your meetings to run longer and business outcomes to be hazy.

Value stream maps can do wonders to fuel continuous improvement. In the software development world, continuous improvement is at the heart of the [continuous paradigm](#) where continuous delivery pipelines deliver products frequently, predictably, and sustainably to customers. When you can release at the speed of ideation, your customers will be happy!

Value stream mapping also helps improve team culture, since productive teams are more engaged and are fun to work with! Since culture, productivity, and savings are just some of the rich dividends, shouldn't value stream mapping be near the top of your single prioritized backlog?

What is Continuous Integration

Software testing for continuous delivery

What Is Continuous Deployment?

Microservices and Microservices Architecture

Bitbucket CI/CD tutorials

Continuous Delivery articles

SHARE THIS ARTICLE



JUNI MUKHERJEE

Juni is a [thought citizen](#) in the DevSecOps space and has made deep investments in the field of Continuous Delivery. She has helped organizations build Continuous Delivery Pipelines, and would love to solve the problems that plague our industry today. She has authored a couple of books.

TUTORIAL

### Continuous Delivery Tutorial

In this guide, we'll see how you can use Bitbucket Pipelines to adopt a

ARTICLE

### Configuration management: definition and benefits

Learn about configuration management and its use in agile CI/CD

continuous delivery workflow. Read on!

[Try this tutorial →](#)

software environments.

[Read this article →](#)

#### CI/CD Topics

Continuous Delivery  
Continuous Integration  
Continuous Deployment  
Pipelines

Software Testing  
Microservices  
Tutorials

Sign up for more CI/CD articles and tutorials.

Email

email@example.com

[Subscribe](#)



Up Next  
[Configuration management: definition and benefits →](#)