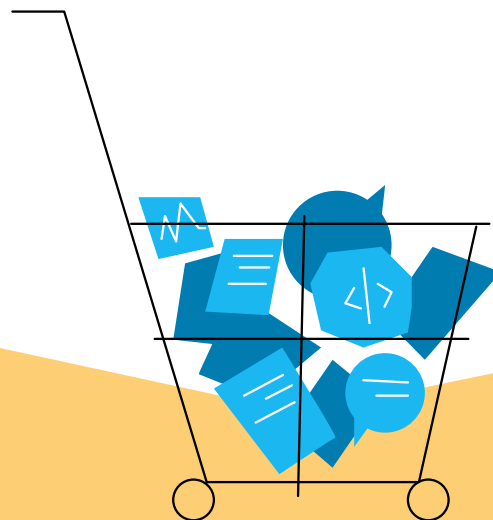


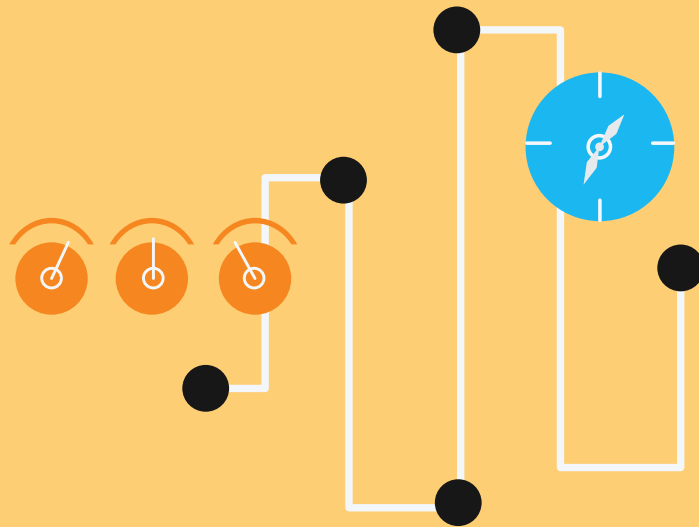
PLUTORA

Everything You Need to Know About Value Stream Management



The world is digitally transforming and expecting cloud and DevOps to enable the desired outcomes for better software delivered faster and safer. But the journeys are arduous and meet many obstacles in the shape of organizational design, technical and cultural debt, problematic leadership, conflicting goals, and lack of key data, metrics, and insights.





Welcome to Value Stream Management (VSM), the way of working that accelerates digital and DevOps journeys through a combination of mapping, visualization, measurement, and governance. **VSM uses real-time data for value stream teams to continually inspect, adapt and optimize performance according to customer feedback on the value they experience.**

In this guide, you'll learn the key essentials to VSM, including the latest trends and expert tips. You'll discover how to implement VSM and access free templates and checklists among the resources.

Table of Contents

What is a Value Stream?	5
What is a Digital Value Stream?	5
What is Value Stream Management?	7
What is a Value Stream Management Platform (VSMP)?	7
The Objectives and Benefits of Value Stream Management	9
A Brief History of Value Stream Management	10
How Value Stream Management Works: An Overview	10
Key Terms in Value Stream Management	13
Key Concepts in Value Stream Management	18
The End-to-End Value Stream	19
Challenges that Value Stream Management Addresses	22
VSM, DevOps, and Agile	23
VSM and ITSM	25
VSM and SAFe	26
Enterprise Value Stream Management	27
VSM and the DevOps Toolchain or CICD Pipeline	28
Case Studies in VSM	30
How to Optimize Value Streams	32
The Value Stream Manager's Job	35
Myths About Value Stream Management	37
Big Trends in Value Stream Management	41
VSM with Plutora	43
Further Resources for VSM	45

What is a Value Stream?

A value stream is anything that delivers a product or a service, or anything that has a customer. A value stream isn't a process, but a collection of processes, steps, or actions that start with ideation and are complete when the value of that idea is realized by the customer. The customer can be either external or internal.

Steps can be value added or non-value added. For something to add value, three things must happen:

1. The step must change the form or function of the product or service
2. The customer must be willing to pay for the change
3. The step must be performed correctly the first time

Some value streams will be 'core' and some will be 'supporting' - they enable the core value streams.

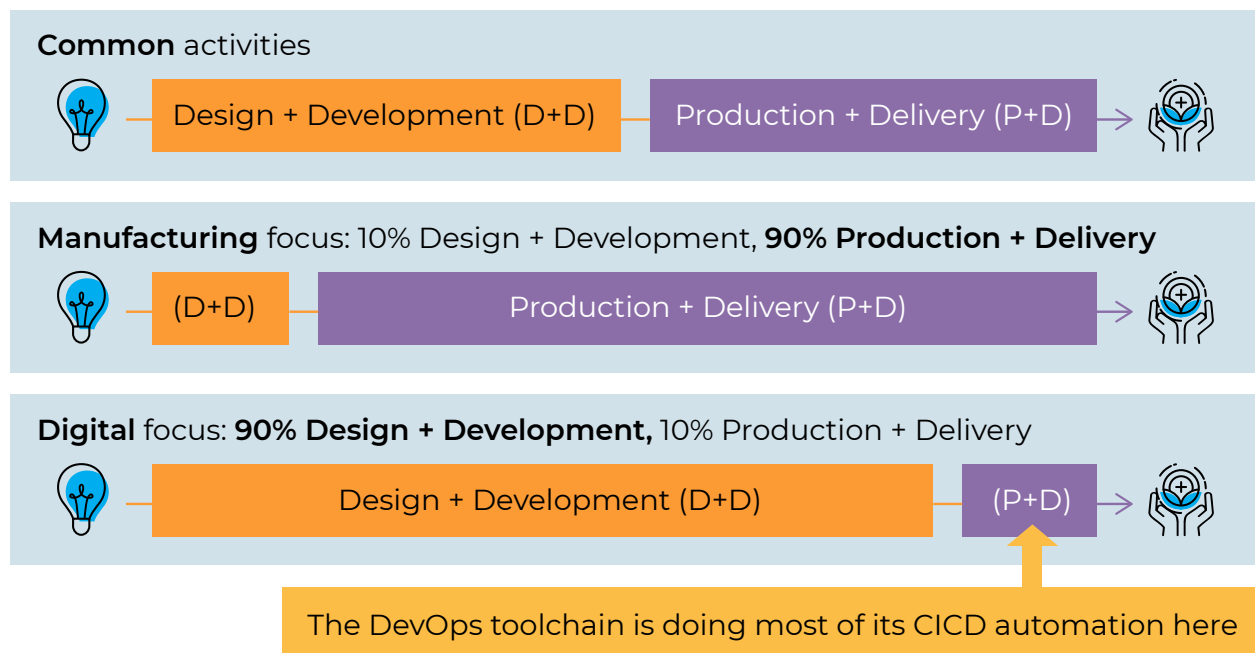
What is a Digital Value Stream?

Digital value streams are concerned with software technology or related infrastructure (cloud or otherwise). Examples of digital value streams include: websites, mobile apps, Software as a Service (SaaS) solutions, Commercial Off The Shelf packages (COTS), APIs, data lakes, DevOps toolchains or Continuous Integration and Continuous Delivery (CI/CD) pipelines and cloud platforms.

Digital value streams are different from manufacturing value streams because in software the product is long-lived, we only make it once, and what's passing through the value stream is enhancements. In manufacturing, we design the product once (or infrequently) and then make it many times according to the same specification.

The two main activities in a value stream are Design and Development (D&D) and Production and Delivery (P&D). In manufacturing, 10% of time is spent on D&D, and 90% on P&D. The opposite is true when we have a digital focus.

Additionally, the use of automation in a **DevOps toolchain** further reduces the time spent in P&D.



Digital value streams are typically core value streams, often supported by supporting value streams such as legal, finance, and procurement. A DevOps toolchain or technology platform value streams, while digital, are typically supporting.

What is Value Stream Management?

Value Stream Management is a way of working that encompasses a number of practices and techniques to organize, map, measure, manage, improve, govern, and accelerate the flow of measurable valuable outcomes to the customer. It makes work visible and in doing so surfaces insights into where delays, waste, and non-value adding work can be removed in order to improve customer experience.

It starts with identifying the value streams themselves, the people involved, and the steps in each value stream. It recognizes that most organizations will have many value streams and that often these value streams are interconnected and have dependencies on one another.

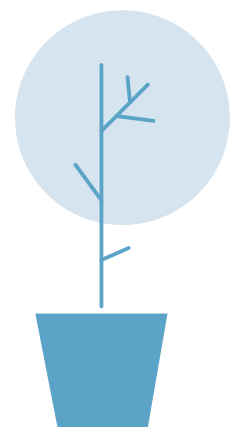
What is a Value Stream Management Platform (VSMP)?

Plutora is a Value Stream Management Platform (VSMP) that enables organizations to simplify building and managing CI/CD pipelines and DevOps toolchains. It minimizes the overhead involved in orchestration, integration, and governance, thus maximizing value by providing visibility, traceability, and observability into the flow of work. VSMPs:

- » Drive business agility
- » Reduce the overhead of managing complex toolchains
- » Maximize flow in digital value streams
- » Support rapid innovation

VSMPs offer several capabilities to achieve these goals:

- » A common data model with data normalization
- » Tools integrations
- » Value stream visualization
- » Real-time metrics into flow and value realization
- » Deep analytics and actionable insights
- » Governance and continuous compliance
- » Simulation and prediction
- » Workflow orchestration
- » Personalization, collaboration, and coordination



The Objectives and Benefits of Value Stream Management

VSM seeks to improve organizational performance by optimizing the flow of value to the customer. It relies on the principle that customers who receive the things they want and need when they want them will be loyal and refer their contacts to the product or service provided.

There is a duality in VSM between flow and realization:

- » **FLOW** is the journey of work from idea to value realization
- » **REALIZATION** is the fulfillment of desired outcomes for customer experience

VSM makes it possible to:

- » Know how long it takes for customers to receive valuable experiences
- » Trace work through the DevOps toolchain and access meaningful data
- » Gain insights into how to accelerate that flow and see improvements
- » Have insight-driven conversations and make insights-based decisions
- » Collaborate on the end-to-end product or service lifecycle
- » Estimate and measure the value delivered from the work done

The key principle of VSM is that if we can satisfy customer demand for value outcomes at optimum speed, their delight results in the consumption of the product and services and referrals that lead to **organizational success**. Harnessing VSM enables businesses to outcompete.

A Brief History of Value Stream Management

VSM's roots are in lean and **value stream mapping**. It's possible to trace manufacturing production lines back to 1450 in Venice's arsenal but it's more common to look to Ford and Toyota in the early to mid-nineteenth century. Toyota's Production System (TPS) included information and materials diagramming intended to visualize and optimize work.

Value Stream Management appeared as a concept in 2000, with work from Peter Hines that recognized there were limitations using mapping alone. In 2018, industry analyst, Forrester, published their first VSM wave recognizing that although VSM itself was not new, the DevOps toolchains and software pipelines that had evolved since 2009 had created a new generation of truly data-driven VSM.

How Value Stream Management Works: An Overview

VSM is a way of working. Not all organizations perform equally and the digital world has created an environment where the choice is to disrupt or be disrupted. To continue to compete, teams need to find ways to

deliver software better, faster, and more safely. They need to balance throughput with stability. They must optimize the delivery of value and the outcomes.

Adopting VSM requires mastery of a number of management techniques from identifying the value streams, organizing around them, and mapping them, connecting DevOps toolchains end-to-end to make the work visible and traceable to continuously inspecting and adapting for insights for further ideation, prioritization, and optimization.

VSM means everyone understands that they are part of the value stream and that the flow of work is more important than the work that they do. If their work is non-value adding, their job is to find ways to remove it entirely or at least minimize the time and effort spent on it, likely through automation. If individuals or teams see themselves causing delays or rework they must find ways to eliminate the wait time or ensure their work can be passed downstream error-free.

Leadership needs to change too. VSM demands a progressive organizational model where authority is distributed and teams are autonomous to avoid delay-inducing handoffs. Leaders need to let go of dominator-hierarchies and evolve to Distributed Autonomous Organizations (DAOs).

Organizations must gain control of their data by:

Architecting and building DevOps toolchains that document and record the flow of work from ideation to value realization.

Connecting these toolchains end-to-end so that work can be traced from the moment it's conceived until its end-of-life; products are long-lived, the impact of enhancements must be measured over time

Using **insights-driven platforms** to find where flow improvements can be made and to measure the impact of optimization experiments through continuous inspection and adaptation.

A VSM Implementation Roadmap:



Key Terms in Value Stream Management

Business Value: In management, an informal term that includes all forms of value that determine the health and well-being of the firm in the long run.

Change Lead Time: A measure of the time from a request for a change to the delivery of the change.

Continuous Delivery Pipeline: A continuous delivery pipeline refers to the series of processes that are performed on product changes in stages. A change is injected at the beginning of the pipeline. A change may be new versions of code, data, or images for applications. Each stage processes the artifacts resulting from the prior stage. The last stage results in deployment to production.

Continuous Delivery Pipeline Stage: Each process in a continuous delivery pipeline. These are not standard. Examples are Design: determine implementation changes; Creation: implement an unintegrated version of design changes; Integration: merge.

Continuous Flow: Smoothly moving people or products from the first step of a process to the last with minimal (or no) buffers between steps.

Current State Map: A form of value stream map that helps you identify how the current process works and where the disconnects are.

Cycle Time: A measure of the time from the start of work to ready for delivery.

Deming Cycle: A four-stage cycle for process management, attributed to W. Edwards Deming. Also called Plan-Do-Check-Act (PDCA).

DevOps Toolchain: The tools needed to support a DevOps continuous development and delivery cycle from idea to value realization.

Digital Transformation: The adoption of digital technology by a company to improve business processes, value for customers, and innovation.

Flow: How people, products, or information move through a process.

Future State Map: A form of value stream map that helps you develop and communicate what the target end state should look like and how to tackle the necessary changes.

Hand Offs: The procedure for transferring the responsibility of a particular task from one individual or team to another.

Hypothesis Backlog: A collection of requirements expressed as experiments.

Insights-Driven: An insight-driven organization embeds analysis, data, and reasoning into the decision-making process, every day.

Key Performance Indicator (KPI): Key performance indicators are the critical indicators of progress toward an intended result, providing a focus for improvement, and on what matters most.

Lean: Production philosophy that focuses on reducing waste and improving the flow of processes to improve overall customer value.

Lean Enterprise: An organization that strategically applies the key ideas behind lean production across the enterprise.

Lean IT: Applying the key ideas behind lean production to the development and management of IT products and services.

Metric: Something that is measured and reported upon to help manage a process, IT service, or activity.

Objectives and Key Results (OKRs): Objectives and key results is a goal-setting framework used by individuals, teams, and organizations to define measurable goals and track their outcomes.

Organization Model: For DevOps and VSM, an approach that is not a dominator hierarchy but instead a Distributed Autonomous Organization (DAO).

Outcome: Intended or actual results.

Outcome Mapping: A methodology for planning, monitoring, and evaluating development initiatives in order to bring about sustainable change.

Processing Time: The period during which one or more inputs are transformed into a finished product by a manufacturing or development procedure.

Project to Product: Changing ways of working from a large batch, waterfall project led approach, to a small batch, agile product (or value stream) approach.

Stream-Aligned Team: A team aligned to a single, valuable stream of work; this might be a single product or service, a single user story, or a single user persona.

The Three Pillars of Empiricism: Three pillars uphold every implementation of empirical process control: transparency, inspection, and adaptation.

Theory of Constraints: Methodology for identifying the most important limiting factor (i.e., constraint) that stands in the way of achieving a goal and then systematically improving that constraint until it is no longer the limiting factor.

Time to Insight Actioned: The time between having an idea, delivering it to the customer, learning, and actioning the insight from that learning.

Time to Learning: The time between conceiving an idea and learning how it was received based on customer feedback.

Time to Market: The period of time between when an idea is conceived and when it is available to customers.

Time to Value: The measure of the time it takes for the business to realize value from a feature or service.

Touch Time: In a lean production system the touch time is the time that the product is actually being worked on, and value is being added.

Value Stream: All of the activities needed to go from a customer request to a delivered product or service.

Value Stream Map: Visually depicts the end-to-end flow of activities from the initial request to value creation for the customer.

Value Stream Mapping: A lean tool that depicts the flow of information, materials, and work across functional silos with an emphasis on quantifying waste, including time and quality.

Value Stream Management Platform: Software that manages value streams.

Wait Time: The amount of time wasted on waiting for work (e.g., waiting for development and test infrastructure, waiting for resources, waiting for management approval).

Waste: Any activity that does not add value to a process, product, or service.

Work in Progress (WIP): Any work that has been started but has not been completed.



Key Concepts in Value Stream Management

Core to VSM is the constant consciousness of the operational health of the value stream. Everyone should know what their value stream is, whether it's core or supporting a core value stream. Value stream teams must define and monitor their own OKRs and KPIs and know what their cycle times and flow rates are on a day-to-day basis.

VSM adoption starts with identifying the value streams and organizing around them and then often the next step is to map the value stream. This is traditionally a manual, human-led process. It's time-consuming but the benefits of learning about the work, each others' roles in it, and starting the VSM journey outweigh the costs of making the exercise happen.

Value Stream Mapping	Value Stream Management
Human-centric, storytelling, and narrative	Tools-centric, science, and insights
Opinion-driven, qualitative	Insights and data-driven, quantitative
Visual collaboration	Visibility and continuous inspection
Walls and post-it notes (virtual!)	Integrations, dashboards and AI
Creates and recreates business case	Ongoing incremental optimization
Highlights waste and assumptions	Automates for continuous compliance
Defines present and future	Records past and present

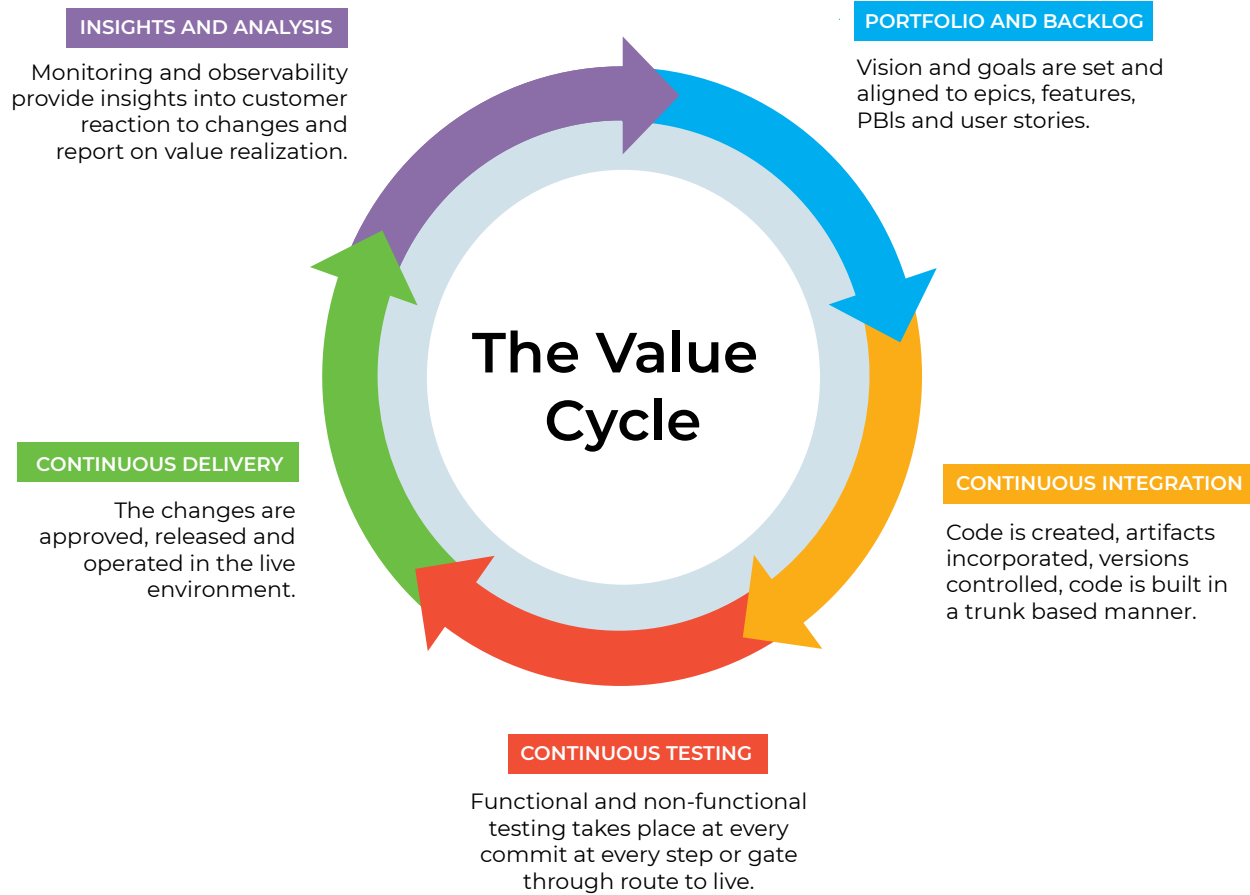
Once the value stream's current state is mapped and the team starts imagining the future state, the real value kicks in as the long-term vision sparks into life and the team builds a hypothesis backlog of the incremental steps that will take them there. But monitoring progress is hard. It can't mean frequent, people-intensive exercises that take people away from their daily work for extended periods of time. Value stream data needs to be available in real-time, all the time. Providing teams with tools that move from data to insights saves them more time, time that they can spend on working on new capabilities that deliver more customer value.

Teams must be able to see where they are expending their effort on:

- » Creating value-adding experiences for customers
- » Improving the delivery system for customers

The End-to-End Value Stream

It's more helpful to think of the value stream as a continuous cycle than a linear set of processes. A value stream starts with ideation and completes, or renews itself, when the value is realized in the hands of the customers. A digital value stream can be divided into five parts at a high level:



1. IDEATION:

Work is recorded in portfolio and product backlog tools here. The primary goals are to ensure new work initiatives align with the organizational purpose and vision, to break the work down into small pieces, refine and prioritize it in line with customer feedback. Collectively, the industry hasn't done a great job of connecting this planning part of the value stream to the rest of it; this changes with value stream management as we make the work items traceable from the moment they are conceived and recorded.

2. INTEGRATION:

Here, the idea is brought to life in the software. It's built in the IDE, probably using artifacts from a repository or new ones that are placed in the repository. Key DevOps principles come into play here. Firstly, **version control** which provides the single source of truth for the teams. VSM's priority is to accelerate and optimize value delivery speed. Defects interrupt flow so anything that eliminates waste like delays caused by errors is part of VSM. Similarly, trunk-based development is designed to continually integrate, build and test software so that 'merge hell' is avoided and issues are picked up and remedied at the earliest opportunity in the cycle (the later they go, the harder, slower, and more expensive they are to fix).

3. VERIFICATION:

Continuous integration by definition includes unit, integration, and **user acceptance testing** but there are non-functional tests (NFTs) to consider before we are ready for delivery too, such as performance and security. Additionally, many teams and organizations have complex route-to-live with many and shared environments which require extra care to manage system and team inter-dependencies, something VSM excels at.

4. DELIVERY:

In an ideal world, our small, autonomous piece of code passes its tests and is automatically deployed to live and available to customers. In most organizations, transitioning from waterfall and project-based ways of working to agile and DevOps ways of working, dependencies

mean releases need to be coordinated across teams and checklists need to be completed. VSM manages all of this and records how long it takes so teams can identify where they can automate, change ways of working, and system architectures to reduce the time it takes to complete this non-value adding activity.

5. OBSERVATION:

Finally, the new code or feature is available to customers. But work doesn't stop here. Now it's time to go back to the user stories in the backlog, ideally written as experiments with an outcome hypothesis. What the teams need is feedback, explicit or data/insights based that tells them how the change has impacted customer experience and how this informs their choices about what to do next. Their decisions, assuming they aren't 'do nothing more' feed into the ideation cycle and round we go again.

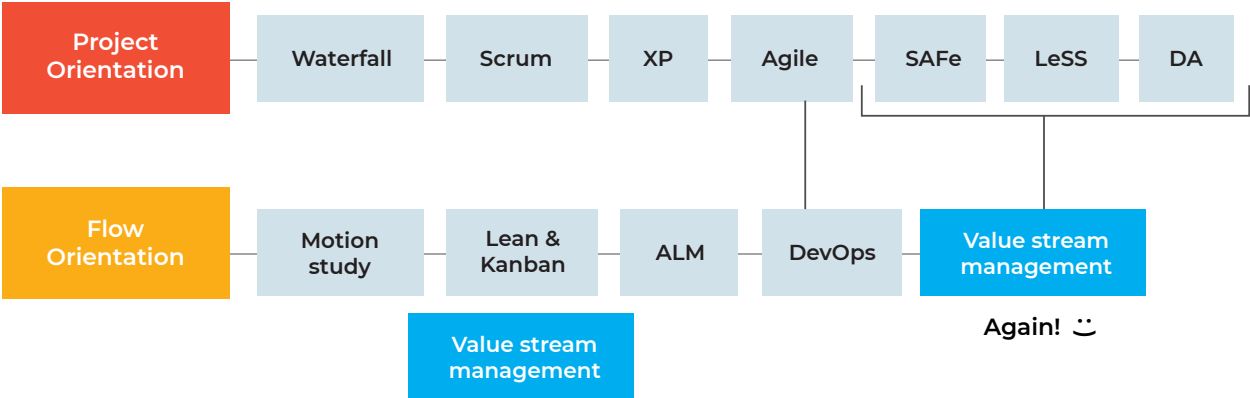
Challenges that Value Stream Management Addresses

At the highest level, it's the ultimate challenge; how to thrive in the digital age. There's a complex system of levers needed to digitally transform, underpinned by the two key enablers: cloud and DevOps. Transitioning from a traditional dominator hierarchy, where work is constructed and delivered around projects in a waterfall manner and the system is riddled with dependencies to a DAO with autonomous, multifunctional teams with loosely coupled systems working in small

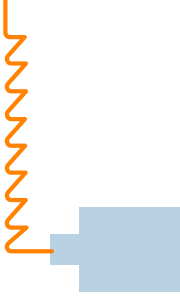
batches and increments is a long and arduous journey. Business as Usual (BAU) has to stop. Time has to be found to unlearn, learn new ways of working, change behaviors, processes, and system architectures.

VSM highlights where to spend that precious time to maximum effect, accelerating the journey. It measures progress on the journey making experiments empirical, giving leaders the confidence positive change is happening. Insights into all value stream steps and traceability between them make the work visible and surface the improvement opportunities. Teams are no longer blind; they know the pace of their work, they know the impact of their efforts. They can make informed decisions about what's next and understand the outcomes from their customers' perspectives.

VSM, DevOps, and Agile



There is no such thing as original thought, we stand on the shoulder of giants, we are all constantly evolving within a groupthink experiment.



What goes around, comes around. All businesses have a common purpose: to serve their customers. Whether that means providing them with the mechanisms to pay their tax online, making and delivering them a new pair of sneakers, or providing them with a service to take them on vacation and accommodate them comfortably in a faraway land. Humans constantly strive to be better, to do better.

There are many ways of working and we name them and describe them so that we can teach each other and learn from each others' experiences. For software, waterfall worked but we were almost always painfully over time and budget. Agile taught us to manage uncertainty by working in smaller batches. DevOps is agile for the IT operations side of the house since developers were doing it anyway. Agile's principles were great but we struggled with them at the enterprise level. So we practiced and documented agile practices at scale: SAFe, LeSS, DA.

In manufacturing, products left the factory, but customer feedback was they wanted them faster and also more perfect, less defective. Lean practices emerged to do just that and we had our first taste of thinking like value streams.

These ways of working crossover, collide, conflate, conflict and compliment. VSM is experiencing a renaissance thanks to the decade we've spent building DevOps toolchains. We automated the value stream without realizing it until we'd done it. VSM is the next generation of DevOps. Or DevOps is the intervention handbook to achieve optimal VSM. Neither matters.

What matters is that businesses work optimally to deliver the value and experiences their customers want and need as fast as they can, without comprising quality, security or governance.

VSM and ITSM

IT4IT	TOGAF	ITIL4
<p>The IT Value Chain content details the series of activities that every IT department performs that add value to a business service or IT service. The IT4IT Reference Architecture breaks these activities down further to a Service Model and the essential functional components and data objects that IT produces or consumes in the IT Value Chain in order to advance the service lifecycle.</p>	<p>Value streams are a result of a breakdown of activities that an organization performs to create the value being exchanged with stakeholders. They provide context and a grouping for business capabilities. They provide valuable stakeholder context into why the organization needs these business capabilities. For every stage in the value chain, the business capabilities that are needed can be identified. The result can be depicted in a new artifact called a value stream stage catalog.</p>	<p>ITIL 4's definition of value stream is "a series of steps an organization undertakes to create and deliver products and services to consumers", all working across the ITIL 4 service value chain (SC). This concept clearly evolves the previous ITIL focus on processes and process outputs by helping people to easily understand how a stream connects processes to services and how operations support business strategy.</p>

A value stream doesn't care whether you are a developer or an IT operations person. It doesn't care if you practice Extreme Programming, scrum, or ITSM. All a value stream cares about is whether you are optimizing flow and understanding your customers' experience.

Traditional processes can carry a lot of weight, in the form of controls, perhaps bureaucracy, that slow work down to the detriment of customer experience. DevOps is partly about bringing agility to IT Operations and then, by its very definition, to ITSM. This manifests in many ways:

- » Non-functional requirements and tests are part of a single product (value stream) backlog
- » The work on checklists (e.g. for release or security) is automated for speed and governance
- » **Change Advisory/Approval Boards (CABs)** are replaced by lightweight, peer-reviewed processes
- » Environment provisioning is automated and self-service using **Infrastructure as Code (IaC)** and cloud
- » DevOps toolchains provide the foundations and capabilities to automate releases and deployments
- » Incident management is dealt with using intelligent swarming and augmented with AIOps

VSM and SAFe

The leading agile at scale framework, **SAFe**, uses value streams as a core organizational model. It's fundamentally built on the concept of value streams and is a flow-based system. Value streams and ARTs are described as "the organizational backbone of a SAFe implementation." It seeks to create a new organizational model that is primarily concerned with the flow of value to the customer - just like DevOps and Value Stream Management. SAFe identifies two types of value streams:

operational and development, the latter being the teams that provide the solutions the business uses to deliver value.

There is a difficulty in DevOps and VSM in treating digital value streams in these two types: DevOps seeks to diminish to the point of exclusion barriers and handoffs between the two disciplines. Defining value streams as operational or development only perpetuates the separation.

Using the two variations of core and supporting is more effective. In a world that has been eaten by software, digital value streams are core. They are the value streams that differentiate against the competition and on which survival is bet. Other value streams exist to support them - whether they are sales and marketing, procurement, finance, or GRC, their role is to ensure the flow and value realization of their associated digital value stream(s) are optimized.

Enterprise Value Stream Management

It's more commonly asked whether VSM can be applied to small, startup businesses than its relevance to large, traditional enterprises. Enterprises are inherently aware that they are constructed of value streams - the work they need to do is identify, name them, arrange around them and work to their tune, daily. Where enterprises come unstuck, and they are painfully aware often of their situation, is that their value streams exist in a multitude, crossing over one another, connected, coupled, and seemingly unravellable in their complexities.

It is a truism of DevOps that dependencies must be broken, not managed. But reaching a fully autonomous, loosely coupled state is not achievable overnight. It's a deeply intertwined system of organizational modeling, system architectures, and cultural and technical debt that needs to be steadily chipped away at.

Dimension	Emerging	Learning	Practicing	Evolving
Insights-Driven	Data manually extracted	Data is aggregated	Tools have been integrated	A single tool connects all parts and automates insights
Dependencies	Aware of dependencies	Managing dependencies	Breaking dependencies	Loosely coupled/ autonomous teams and systems
DevOps Toolchain	Building continuous integration	Using continuous delivery	Architected from idea to value realization	Work is traceable around entire cycle automated value stream map
Metrics	Incident rate, change fail rate	Deployment frequency, MTTR	Lead time, cycle time	Flow velocity, efficiency, value realized
Organizational	Starting to use value stream mapping	Naming value streams, some roles	Teams directed around value streams and customer journeys	All teams organized around value streams, dedicated roles

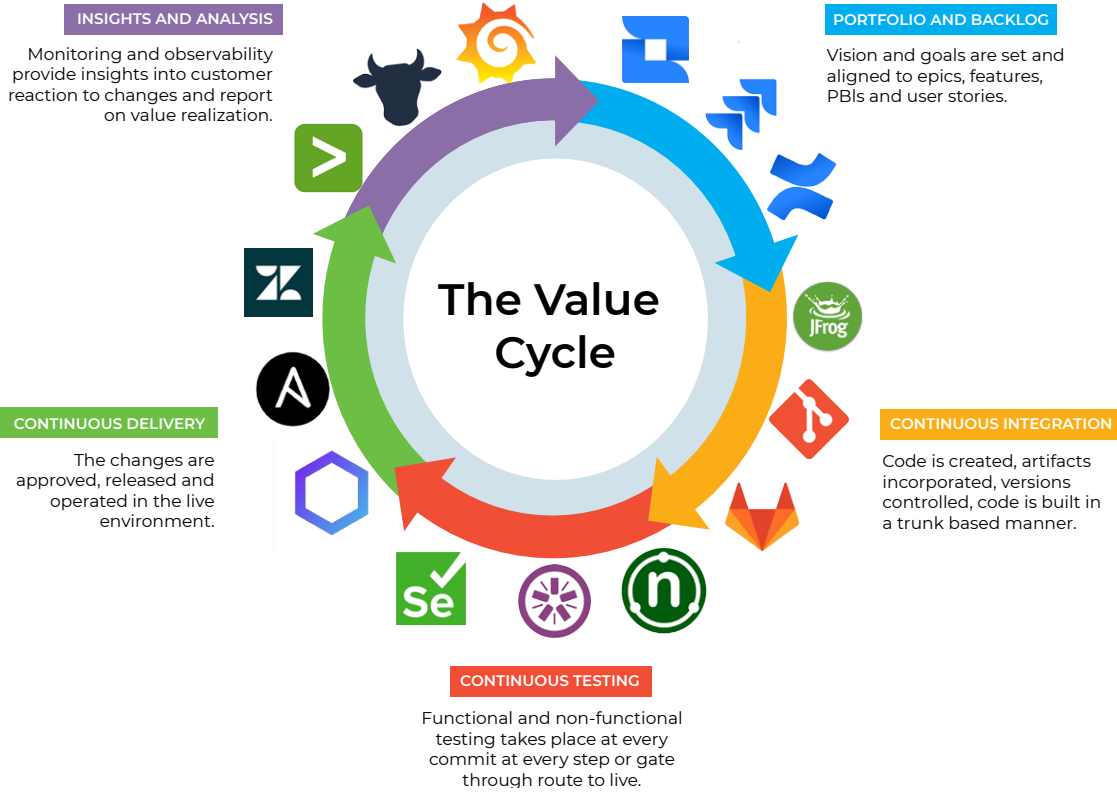
VSM and the DevOps Toolchain or CI/CD Pipeline

VSM is about people and process and tools. DevOps roared into life at the backend of 2009 and is a way of working movement driven by demand to balance and improve throughput and stability. It's about culture as much as it's about automation but it's what we've learned about building DevOps toolchains that really changes the VSM game.

We started with version control and evolved to continuous integration which allowed us to continually deliver. **CI/CD pipelines** became

commonplace. But they didn't handle the end-to-end software development and delivery processes.

For that, we needed DevOps toolchains that incorporated the definitions of work in the form of portfolio management and product backlogs. And also the tools that manage software post-release; the service desk tools and the **monitoring and observability** that provided the feedback to keep driving improvements and enhancements around the cycle. Now, we have data points at every step in the digital value stream.



Challenges remained though:

- » How to get at the data and make it useful
- » How to integrate the heterogeneous tools to be able to observe flow

Enter VSM. Experiments with DevOps dashboards were limited in usefulness since they effectively scraped and presented the data points without offering traceability through the end-to-end toolchain. They also weren't capable of augmenting human analysis with insights. Data is not useful until an actionable insight has had a learning outcome.

VSM platforms like Plutora do things differently. They:

- » Integrate all the tools in the end-to-end software development and delivery lifecycle (the digital value stream) making it possible to trace work items from their inception to realization.
- » Can do this regardless of the heterogeneous tools in a pipeline and variations across multiple DevOps toolchains providing enterprise level visibility.
- » Automate the value stream map and the key data we'd expect like processing, wait and cycle times but also extract insights into delays, waste and flow acceleration opportunities.

Case Studies in VSM

This all sounds great in theory, but let's look at this in practice. A great place to start is New York health insurance provider, **Healthfirst**.

Like many organizations, Healthfirst was already in a transition to the progressive ways of working exemplified by agile. Their CICD and DevOps capabilities were earlier in their evolution. Like many organizations, the Covid pandemic demanded a rapid pivot on their existing business model. Part of this was rapid delivery of a new mobile app.

As Healthfirst began to adopt DevOps ways of working, they partnered with Plutora to help them improve their release quality while accelerating deployments. Having visibility into the teams' work in their value streams meant that they were able to manage their way through a period of heightened uncertainty and make rapid operating model changes because:

- » They were able to automate governance and trust teams to release faster
- » And consequently make the CAB process lightweight to avoid delays
- » Plutora meant teams could see and coordinate interdependent release processes
- » Which meant they continued to reduce risks
- » Teams could accelerate DevOps toolchain implementation as they learned from each others' patterns
- » They could see and throttle release scope creep further reducing risk

Healthfirst is practicing SAFe for **agile at scale** and Plutora supports their goals for managing working practices at an enterprise level. It provides a converged view of the complex, multiple change records

that exist in their service desk solution that relate to multiple releases across all their applications. This provides complete visibility into the status of the release, something that previously several systems would have to be queried to piece together the understanding.

Ultimately, Healthfirst aiming for a state where they are able to see the flow of work through each value stream and gain insights for further optimization whilst continuing to balance speed and risk.

How to Optimize Value Streams

Type of Waste	How the Waste Manifests
Defects	Time spent fixing bugs and paying down technical debt
Overproduction	When capacity is spare in teams and they are starved of requirements
Waiting	Teams are siloed and handoffs cause delays
Non-Utilized Talent	People are not empowered to choose their work and make decisions
Transportation	Unnecessary emails, irrelevant reporting
Inventory	Tools being acquired and not used effectively
Motion	Unnecessary steps in processes
Extra-Processing	Backlog items delivered that have no value outcome

It's only possible to know for sure that improvements are happening if they are measured. Agile teaches an empirical and experimental approach where we imagine the outcome of our actions in a hypothesis and visit the change at a specific time (or times) in the future to see what actually happened.

When optimizing flow, the lean waste acronym '**DOWNTIME**' is useful to follow:

Example hypotheses formats used for optimizing VSM flow are:

- » If we do <this> to remove <this identified waste> in our value stream, we think that our cycle time will improve by <time measure> on <this date>
- » When we automate <this non-value adding task> it will reduce the processing time from <time measure> to <time measure> in this <timescale>
- » By improving our **flow time** by <% or multiple> we will be able to deliver <increased volume> more enhancements to our customers, worth <value> in outcomes

Improvements to value stream flow are made by removing:

- » Waste
- » Delays within steps and between steps (wait time)
- » Non-value adding activities

Examples of interventions to optimize digital value streams based on good DevOps practices include:

- » Shortening the time between generating an idea or receiving it from customer feedback to starting work on it through small batch, autonomous product ownership

- » Using continuous integration and associated test automation to merge and accelerate software delivery and validation steps
- » Practicing continuous delivery to accelerate frequency and volume of value outcomes to customers
- » Shortening the Route to Live (RtL) using cloud, IaC and self-service environment provisioning practices
- » Using limited blast radius and dark launch techniques such as **blue/green deployments** and **canary releases** to deliver faster with less risk
- » Automating the deploy and release processes using standard configurations and processes to accelerate the step and mitigate interdependency risk
- » Using cloud-native architectural techniques such as **microservices** to create loosely coupled environments where system components can be independently built, tested and deployed
- » Replacing heavyweight Change Approval/Advisory Boards (CABs) with automated checklists built into the CI/CD pipeline/DevOps toolchain and lightweight peer-based reviews
- » Replacing heavyweight incident management processes that cause exacerbate unplanned work with swift, intelligent swarming
- » Employing **DevSecOps** practices and tools to reduce delays caused by wait time on security teams

- » Organizing around value streams as multifunctional, autonomous teams to reduce handoffs

In order to measure the impact of these interventions on flow, teams must have realtime data on processing, wait and cycle time available from their preferred toolchain. To achieve that goal, they'll need a VSM tool, like Plutora.

The Value Stream Manager's Job

Fully embracing VSM requires the introduction of value stream-specific roles. The most essential role to assign is that of the **Value Stream Manager**. Without it, there is no fulcrum of accountability for the value stream's operation. DevOps and therefore VSM promote multifunctional, autonomous teams and also multifunctional people; roles are valued more than job titles. This role then can be assigned to an individual who also carries other roles, for example, Product Owner or Scrum Master.

The duties of the Value Stream Manager include:

- » Defining the product family by conducting product routing analysis and appropriate groupings
- » Ensuring that a current state value stream map is created of the end-to-end value stream
- » Conducting fact-based analysis of the current state map

- » Preparing future state maps showing what the value stream could look like in the long term
- » Preparing target state maps that use lean techniques to eliminate waste and improve process value in the short to mid-term
- » Creating a plan to achieve the future state along with OKRs and KPIs
- » Ensuring work items are in the value stream backlog and are refined, prioritized and delivered
- » Implementing a VSMP to automate the generation of maps and improvements
- » Definition of metrics and enablement of continuous inspection and adaptation
- » Leading the continuous implementation of the plan
- » Leading and mobilizing the people inside and outside of the value stream to enable the required changes
- » Leading the day-to-day activities within the value stream to ensure that current commitments are achieved while improvements are being made
- » Increasing the ratio of value adding activities to non-value adding activities

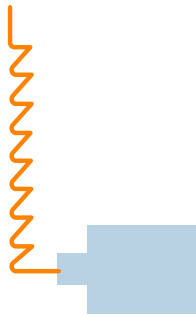
- » Eliminating waste through the end-to-end value stream from idea to realization
- » Ensuring optimal customer experience, gathering and acting on feedback
- » Measuring flow and improvements to cycle time through experiments
- » Managing and breaking dependencies with other value streams
- » Treating the value stream as a microbusiness and managing the P&L

Myths About Value Stream Management

Complex systems of thought and ways of working are always misunderstood and misinterpreted by some of the marketplace, particularly when they exist in their early stages of evolution and VSM is no different. Myths can be useful as they offer a framework for positive definition, discussion, and challenges around the principles and practices. Here are some of the common myths around VSM and their corrections:

Myth 1: Value Stream Management is Value Stream Mapping.

Value Stream Mapping is currently more widely understood and practiced than full Value Stream Management and the two are frequently conflated. Mapping is a sub-discipline of the group of practices that comprise Value Stream Management and an essential



component but in digital value streams, it must be combined with organizational change and continuous inspection and adaptation of data from DevOps toolchains to fully deliver on its promises of organizational performance improvement.

Myth 2: Value Stream Management is only for large enterprises.

Anything that has a customer or delivers a product or service has a series of steps or value streams that are followed and completed for the successful delivery of a customer experience that has a value outcome. Even if a business is brand-new or consists of a single value stream, there are benefits to be gained from optimizing flow and feedback to and from the customer.

Myth 3: Value Stream Management and DevOps are incompatible.

The concern derives from the reality that DevOps toolchains are already hugely complex in their inherent heterogeneity and their cross-organization variability. The requirement to maximize the data available from the DevOps toolchain requires the addition of another component, the VSMP, to access the insights essential to optimization. But a VSMP connects all the tools in a way that is unachievable, or incredibly time-consuming, to do manually and makes the insights available so that teams don't have to do this work themselves. VSM is either next-generation DevOps or DevOps is the intervention handbook that powers VSM. Either way, they don't conflict but the combination of the two is the most powerful way to delight customers through the fast flow of enhancements that deliver measurable value outcomes.

Myth 4: Value Stream Management creates a blame culture and gives a new stick to management.

Actually, VSM, like DevOps, promotes the cultural imperative of the Distributed Autonomous Organization (DAO). Traditional management doesn't exist - the value stream is managed, not the people. Leaders' roles are to help teams self-discover improvements. Value stream teams define their own OKRs and KPIs in line with the organizational vision and they are accountable for continually inspecting, adapting, measuring progress and surfacing and sharing success to make local discoveries global improvements.

Myth 5: Software developers don't care about VSM.

Every role in the value stream is required to understand their step in the end-to-end value stream. In a digital value stream, software developers are accountable for the most value-adding step. Developers care enormously about customer experience - it's a sign of their pride in their work and their craft. VSM, like DevOps, cares about employee engagement and burnout. It knows that autonomy, mastery, and purpose are key to happiness and productivity for knowledge workers. Aligning individual purpose with organizational purpose is one of the biggest challenges businesses face today. Providing developers with real-time feedback on the impact of their work on the people that use their creations is a massive enabler for the continual strive for optimal customer experience. And great customer experience equals high organizational performance. Joy is found in progress and success.

Myth 6: Value is indefinable and unmeasurable.

It's true that value is in the eye of the beholder, but there are many ways to see and measure customer experience, understand value outcomes of changes to digital products and services and learn from this feedback. Lagging indicators like sales, revenue, and profit and intentional indicators like reviews and referrals, conversions, retention, customer journey times, basket size, and bounce rates for example. Value Stream Managers are accountable for the P&L of their value stream through continuous inspection of the impact of improvement experiments on customer experience and behavior and adjusting the costs of sale and production in line with income.

Myth 7: We have too many interconnected value streams to make managing them possible.

Actually, this is one of VSM's strengths; making the disparate value streams visible, the connections, contentions, and conflicts between them. VSM makes it possible to manage dependencies and shows where to focus effort on breaking them.

Myth 8: DevOps is enough.

Research shows that DevOps evolution is stuttering in many organizations and VSM is the key to unsticking the enormous change required for digital transformation and enabled by DevOps and cloud. Teams struggle to know where to make improvements and can't see their work or their progress. Leadership lack anything other than

narrative to see how their efforts and investments are having an effect. VSMPs make all this visible and DevOps delivers on its promises.

Myth 9: I can do VSM without a VSMP.

Sure you can, but why miss out on using all that data you've already invested in collecting and augmenting your teams' cognitive load by providing them with insights that will improve organizational performance? Automating your value streams, their maps, and being able to continually inspect and adapt them is the optimum improvement position.

Myth 10: We don't have time to implement VSM.

The real question is, can you afford to not implement VSM? Research shows that VSM adoption correlates with organizational performance so businesses that don't harness these ways of working will lose pace with those that do. VSM is key to business survival in this chapter of the digital age.

Big Trends in Value Stream Management

It's the emergence of VSMPs like Plutora that are driving the resurgence of VSM. Automation makes life easier; it saves vast amounts of human effort, reduces unplanned work and rework, provides insights, makes things measurable, and builds on existing investments. The key trend in VSM is the adoption of the platforms that supercharge its goals and efforts.

Value Stream Mapping is a well-established component of VSM and rapidly increasing as both a service that's delivered to organizations by consultants and a discipline that organizations are enabling teams to master themselves. There is great variation in approaches across the industry though, so expect to see some standardization, good practices, tools, and templates emerge to help accelerate this step and embed the habit using VSMPs.

DevOps metrics such as deployment frequency, lead time (from code commit), change fail rate, and MTTR pushed us to where we are now, but flow metrics are emerging as the next phase of our evolution. Cycle time (from an idea) and flow time and efficiency are key as are measuring the ratios of value to non-value adding to improvement work in our daily systems.

Increased understanding of value metrics themselves, what the customer is actually experiencing, and what the team can learn and action from these is the other big trend. Research analysts have suggested renaming the AIOps segment to AIPA (AI Predictive Analytics) in part to acknowledge the increasing importance of customer experience resulting in an increase in teams using observability solutions to understand value, not just resolve incidents.



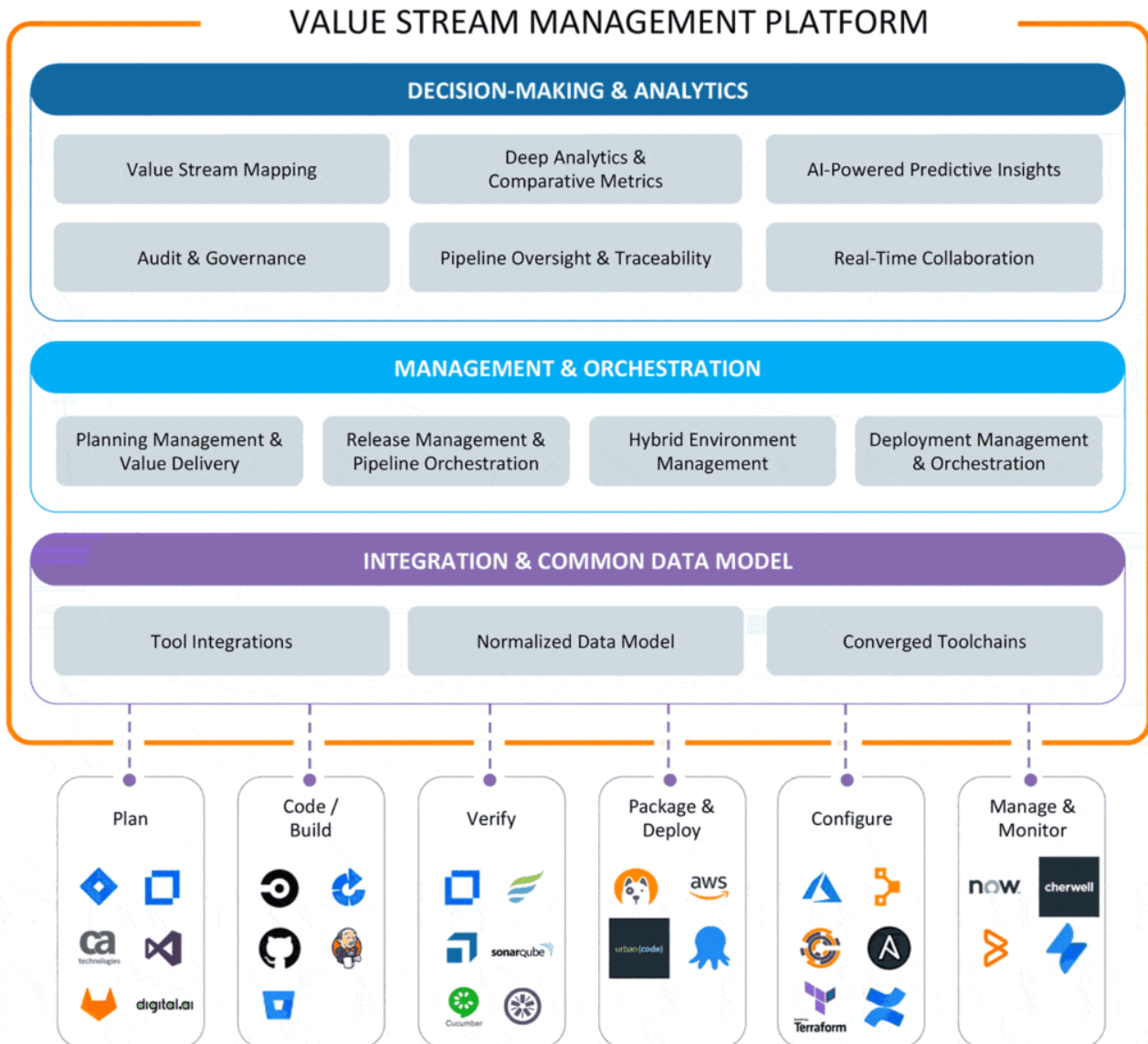
VSM with Plutora

- » Plutora is a Value Stream Management Platform (VSM). It supports these VSM objectives:
- » Break down operational silos by connecting multiple processes, teams, and tools
- » Leverage real-time metrics
- » Enable cross-team collaboration
- » Ensure governance is embedded into SDLC processes
- » Coordinate and automate workflows

It does this by providing:

- » The ability to integrate with tools in the CICD pipeline and DevOps toolchain
- » A data normalization engine
- » Visualization of value streams and maps
- » Metrics, KPIs, and OKRs - realtime
- » Deep analytics, AI, and actionable insights

- » Governance and continuous compliance
- » Workflow orchestration
- » Personalization, collaboration, and coordination





Further Resources for VSM

- » Book: [Mastering Software Delivery with Value Stream Management](#)
- » Blog: [What is Value Stream Management and Why is it So Important?](#)
- » Webinar: [VSM and DevOps Predictions 2022](#)
- » [VSM Consortium](#)

Authors



Helen Beal | *DevOps Coach*

Helen is a DevOps and Ways of Working coach, Chief Ambassador at DevOps Institute and an ambassador for the Continuous Delivery Foundation. She is the Chair of the VSM Consortium and provides strategic advisory services. She is also an analyst at Accelerated Strategies Group. She regularly appears in TechBeacon's DevOps Top100 lists and was recognized as the Top DevOps Evangelist 2020 in the DevOps Dozen awards.



Jeff Keyes | *VP of Product Marketing and Strategy at Plutora*

Jeff is a DevOps Institute Ambassador, a founding board member of the VSM Consortium, and a product evangelist passionate about software development and enabling organizations to reach their product delivery goals. With a deep background in designing and developing software products including time at Microsoft, he is now focused on growing startups and enterprises alike with value stream management.