Whitepaper: Success with enterprise DevOps

Edition 202212

About the authors

Koichiro (Luke) Toda
President of Strategic Staff Services Corporation
Director of TPS Certificate Institution
Contact: lktoda@ask3s.net

Nobuyuki Mitsui
CTO of Strategic Staff Services Corporation
Contact: nmitsui@ask3s.net

Content

Success with enterprise DevOps

Enabling IT services to support the business using DevOps.

Introduction

We, the authors, have been providing coaching services for Agile development and Toyota/Total Management System (TMS) which is the core of the Toyota Production System (TPS). Based on our experience, we believe that DevOps can strongly support the business. This whitepaper uses a case study, which helps to explain the use of a DevOps framework.

DevOps can be used in more areas than just supporting IT. DevOps can also be used to support the business strategy and to improve business processes.

There are many books about DevOps. Three useful books to understand what DevOps is, are:

- **The Phoenix Project** (Kim et al., 2018) is written from the IT manager's perspective.
- **Continuous Delivery** (Humble & Farley, 2010) is written from the developer and project manager's perspective.
- **DevOps a software architect's perspective** (Bass et al., 2015) is written from the architect's perspective.

This whitepaper is written from a business process perspective because the intention of DevOps is to establish a software/IT services supply chain that supports the business and to manage the process to reach maturity.

A journey to DevOps – case study

Our client provides high-grade web conversion services: from fixed workstation to mobile devices The journey to DevOps started in 2009, after they had successfully implemented Agile, Scrum, and XP methodologies.

Their Scrum team is developing and releasing software much more quickly than earlier. However, the business director had his concerns that the business speed could not be improved even when the development time was halved. At first, it seemed that the development process was the bottleneck. Upon investigation, it was found instead that the business process could be improved.

The concept of TMS was implemented in the entire business process: from business strategy and planning, to customer services. This helped to:

- establish a streamlined business operation and
- shorten delivery lead times for business services using the DevOps concept.

This project was successfully completed in 2012. The whole process was realigned from end to end. Collaboration across the business was set up, using visual control, one-piece flow, weekly synchronization of processes, daily feedback loops and Kaizen[1]. Managers, administrators, sales staff, designers, developers, operators, and customer support staff are one team and share all the business information on visual boards.

After implementation, the business results were dramatically improved:

- shorter delivery lead times
- increased sales volumes
- increased profitability
- improved motivation of staff

This is the true benefit of DevOps. The DevOps framework should support business outcomes directly and not only through collaboration of Development and Operations for IT services. Enterprises are using IT services to support and improve their business.

The use of DevOps should be evaluated by business outcomes, not by an IT project scope and IT outcomes.

[1] See Glossary

What is DevOps for the enterprise system?

There are many books about DevOps, but unfortunately, most are describing the use of DevOps for web- and product development. There is very little information about the use of DevOps for the enterprise system.

The enterprise holds both the system of engagement (SoE) and the system of record (SoR).

- The SoE is focused on speed.
- The SoR is focused on business continuity.

The question is how the SoR can adapt quickly to changes to the SoE. Gartner calls this the Bimodal challenge[2].

The SoR in most enterprises is struggling with legacy applications/systems and can be amended with DevOps by building streamlined processes with just-in-time (JIT) concepts.

DevOps is not a single tool, methodology, skill set, or organization structure. DevOps is a framework combining all of these to help organizations establish streamlined processes, to enable the business to operate faster, and to respond to change more quickly. DevOps can also enable enterprise maturity by using Deming's Plan-Do-Check-Act cycle (PDCA).

Enterprise DevOps is not only an enhancement of Agile development and continuous delivery but also of IT service management and application management, to enable growth of the business and to maintain business continuity.

What is the goal of DevOps?

The goal of DevOps is to establish streamlined, just-in-time (JIT) business processes. DevOps aims to maximize business outcomes, such as increasing sales and profitability, enhancing business speed, or minimizing operating costs.

DevOps means establishing the IT service supply chain in the business in the same way as the supply chain for other products is embedded within the business. It is a paradigm shift; from software delivery to providing IT services.

From an architecture perspective, DevOps must establish an automated quick deployment system. There are many methodologies and tools which can be used. DevOps does not have a template for implementation. Each organization will have to think about and build their own DevOps process to improve the business. Therefore, understanding the concepts of DevOps is important to carry out the right processes efficiently.

[2] *Definition of Bimodal - Gartner Information Technology Glossary.* (n.d.). Gartner. https://www.gartner.com/en/information-technology/glossary/bimodal

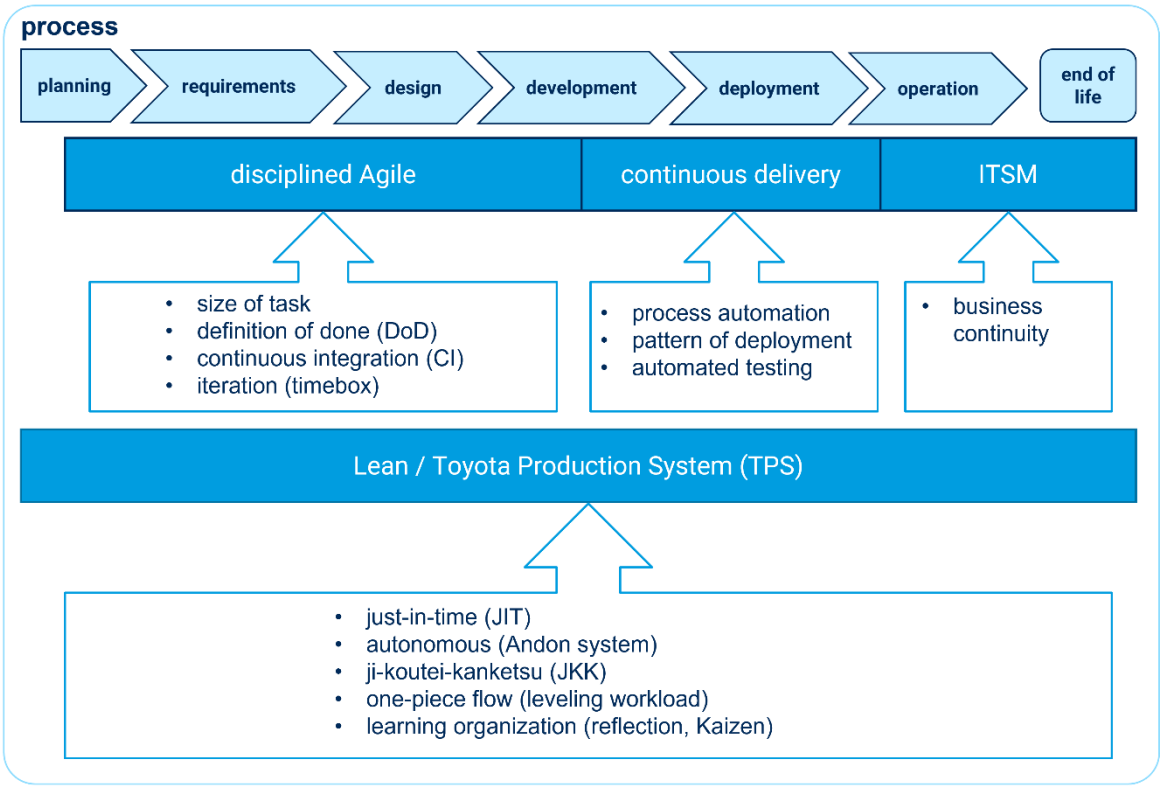
DevOps body of knowledge

When implementing DevOps, there are many sources of knowledge, methodologies, practices, and tools to choose from. DevOps consists of 3 pillars and a foundation.

The pillars are:
- Disciplined Agile
- Continuous delivery
- IT service management (ITSM)

The foundation is:
- Lean / Toyota Production System (TPS)

Figure 1 DevOps body of knowledge



Disciplined Agile

A disciplined Agile Development team is key to the success of a DevOps implementation. Disciplined Agile means:

- Stabilized velocity
- Adaptability to change
- Release of high quality, bug free code

A more frequent and faster release cycle of IT services to respond to business change depends on the speed of development. The quality of work should be most important. This can be established by splitting work into small tasks.

Continuous delivery

Continuous delivery is the automated implementation of the application build, deploy, test and release processes. A key focus is on testing, for example through acceptance testing and performance testing. TPI Next can be useful for the maturity of this process.

Every organization will be different in the implementation of their deployment pipeline, depending on their value stream for releasing software. A key success factor (KSF) for continuous delivery is establishing a single deployment pipeline for IT services.

IT service management (ITSM)

Continuous or high availability of IT services is critical to the survival of the business as a whole, because technology is a core component of most business processes. This is achieved by introducing risk mitigation measures and recovery options. Like all elements of IT service management, successful implementation of the service continuity process can only be achieved with senior management commitment and the support of all members of the organization. Ongoing maintenance of the recovery capability is essential if it is to remain effective. Service continuity is an essential part of the warranty (fitness for purpose) of a service. If service continuity cannot be maintained and/or restored in accordance with the requirements of the business, then the business will not be able to exploit the value that has been promised.

Traditional IT service management (ITSM) best practice such as ITIL looks heavyweight and not suited for the quick processes of DevOps. It is necessary to think about how to reduce management workload.

It is necessary to realign ITSM for DevOps, creating lightweight ITSM which is strictly focused on business continuity with a set of minimum required information (MRI). The MRI set for each organization depends on their business.3

Toyota Production System (TPS) and Lean as foundation

Building a streamlined supply chain of IT services is complicated because there are many elements and it is necessary to change our mindset, away from the familiar development cycle and its methodologies. The concepts of TPS, which includes just-in-time (JIT) and automation, can help. JIT means building a streamlined supply chain with one-piece flow. Automation means automating as much as possible and stopping the entire process when a defect occurs.

The process must be designed for and staff educated in these two concepts. The other key issue is the management cycle of development and operation. This must change to working in an Agile way, including synchronization between development and operation on a weekly or daily basis.

Ji-Koutei-Kanketsu (JKK)

The concept of Ji-Koutei-Kanketsu (JKK), which means 100% completion of an item, helps to maintain high quality of work. The definition of done (DoD) or completion must be defined clearly for everybody. The product owner may change his or her mission from not only managing product backlogs but also planning the operating cost of the IT service, which was actually done by Toyota's Chief Engineers.

[3] See Whitepaper Lightweight ITSM for DevOps

DevOps team roles

It is recommended that a DevOps team is set up in the organization to commit to business continuity of the IT service. Small DevOps teams can be composed according to Amazon's "two pizzas rule": a team should be small enough to be fed with two pizzas!

The team roles are:

- Process Master
- Service Master
- DevOps engineer
- Gatekeeper/release coordinator
- Reliability engineer (optional)
- Developers
- Operations team members

Process Master

The Process Master leads and facilitates the team. This role is the very similar to the Scrum Master role in Scrum. They implement visual control across the entire process and have a strong focus on establishing a streamlined process with one-piece flow. Visual control means "Does everyone easily understand the situation by just looking at the boards[4]?" The board does not show the status. It shows which problems have occurred and which have not.

Required experience: as Scrum Master or as Agile Project Leader

Service Master

The Service Master is responsible for providing IT services just-in-time (JIT). This role is very similar to a Product Owner role in Scrum. The Product Owner manages product backlogs and prioritizes items. A Service Master has the additional responsibility of cost planning for the IT service.

Required experience: as Scrum Product Owner or as Service Owner

DevOps engineer

The DevOps engineer has a mission to improve and maintain automated processes. The engineer will examine the whole automated process and its tools. Many tools are required in the DevOps process.

Required experience: in Development and with tools

Gatekeeper / release coordinator

Responsible for monitoring the operational status and progress of the next release of the IT service. Makes go/no go decisions regarding deployment, according to criteria like security, compliance, readiness of the Operations team and their process views.

Required experience: in IT service management and Operations

[4] The 'board' can be any type of board: a Scrum board, a Kanban board, a digital board, a physical card wall, or any other means of visualizing the status of the work.

Reliability engineer (Optional)

Monitors the services in the deployment process and deals with problems with the service during its execution. Monitors the process status to ensures that the Development team are following the rules of CI (Continuous Integration) and CD (Continuous Delivery). Monitors and manages the flow of the complex build pipeline. Has a mission to improve the test process.

Required experience: in Testing, Tools, and Quality Assurance

Developers

One of the key success factors for DevOps is building a disciplined agile team. Disciplined agile teams commit to meet release plans and quality in sustainable pace.

Required experience: in Development and with Agile

Operations team members

Adopt lightweight ITSM and support the design, implementation, operation, and improvement of these services within the context of an overall strategy. Apply “Kaizen in Advance” which is a practice of Kaizen in TPS.

Required experience: in Operations and with Kaizen

Organization

It is useful to organize the DevOps team in a service management office to support the Service Master. There are two types of organizational structure.

- **Flat organization** for small organizations.
  - This is the basic structure for a small team.
- **Matrix organization** for large and complex organizations.
  - Pooling experts and assigning them as a team to a Service Master. The idea for this matrix organization came from the chief engineer in Toyota.

Figure 2 Flat organization

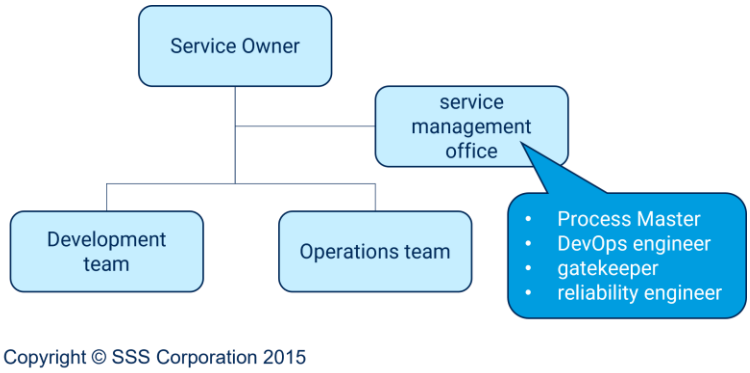
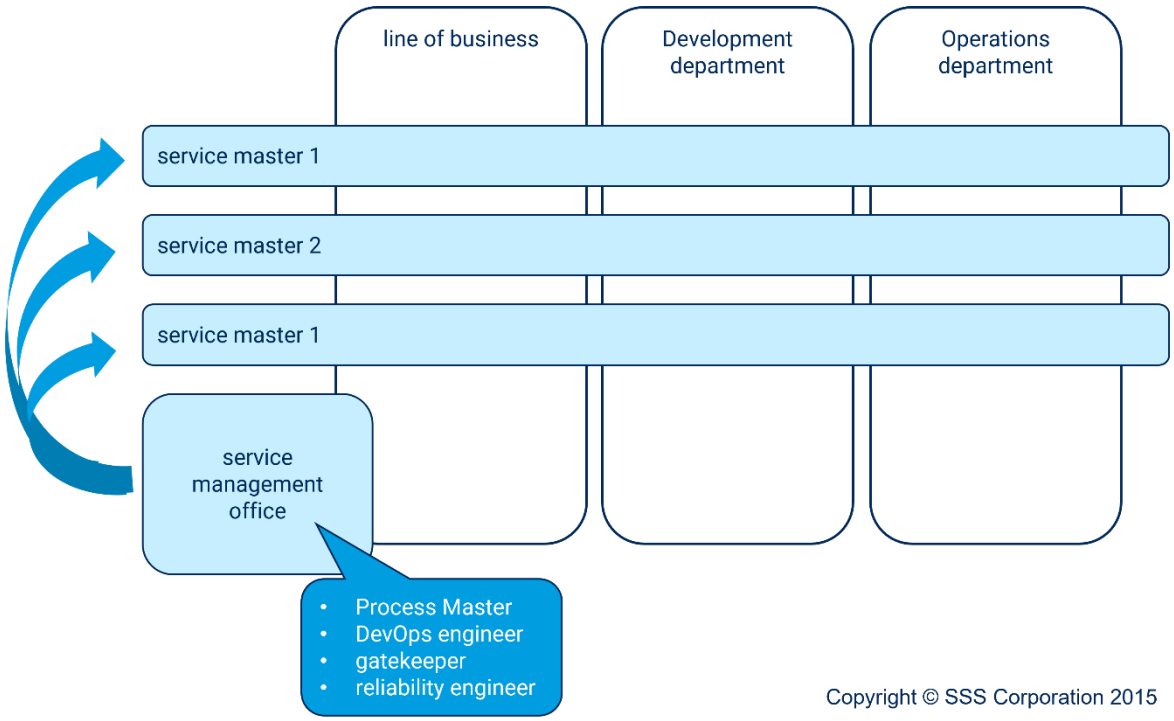


Figure 3 Matrix organization



DevOps process

To build up streamlined processes, ji-kotei-kanketsu (JKK) is the most effective method to guide the behavior of the DevOps team. JKK is a way of working for quality, which means clear understanding of the goals, understanding the right way to work, getting the work right for 100% completion and then maintaining the required quality without inspections.

Business strategy and planning

IT services should have a close relationship with the business strategy and business plan. The Service Master should attend business planning sessions and make recommendations on how to gain business advantages from the IT services.

Marketing and sales

The Service Master should discuss with marketing how to gain advantages from the created IT services.

The Service Master:

- identifies customers of IT services
- gathers requirements that will generate business value
- determines a good time frame

Administration

The Process Master explores how to visualize the whole process. One method is to use Obeya, which can be set up for the whole process. Obeya is a 'war room' which serves two purposes:

- information management
- on-the-spot decision making

There are many visual management tools in it. Team members can quickly see where they are in every aspect of the program.

When a cross-functional team works together, the Obeya system:

- enables fast and accurate decision making
- improves communication
- maintains alignment
- speeds up information gathering
- creates a strong sense of team integration

Project planning

The Service Master organizes the service management office (SMO) and defines ground rules for the team. A Service Master creates the vision and goal of the project, determines the value of the project, and then composes the DevOps team.

The run-time infrastructure is defined at the project planning stage. A value stream map (VSM) of the whole process is designed at this stage as well.

Requirements and design

During the requirements and design phase, the following happens.

- The Service Master defines the product backlog and the priorities.
- Then DevOps team members use the product backlog to define stories.
    - **User story**: role, function, business value/reason, and conditions of operation.
    - **Test story**: acceptance test cases and service acceptance criteria.
    - **Operations story**: set priorities of IT services and conditions of operation for business continuity. Create service level and operational level agreements.
- The DevOps engineer and Operations team together define the transition, test and development infrastructure.
- The Development team creates release and iteration plans.
- The gatekeeper investigates the compliance and regulatory requirements for the IT services.
- The reliability engineer defines the testing methodology and test cases.

Development

The next stage is the actual development of the services. Scrum is the most applicable framework at this stage. The Development team must commit to release plans and then work using Disciplined Agile approach. The period of each iteration (sprint) is determined in accordance with business needs.

From a quality point of view, extreme programming (XP) practices are effective:

- Pair-programming
- Test-driven development (TDD)
- Refactoring
- 10 minute builds

The Development stage benefits from continuous integration as well.

Deployment

After continuous integration during development, the automated processes for the acceptance test, performance test, and deployment should start.

- The DevOps engineer should build a single automated deployment pipeline using one-piece flow.
- The reliability engineer and DevOps engineer collaborate to improve the testing process.
- The gatekeeper monitors progress across the process and makes the go/no go decision about going live.
- The Operations team studies how to maintain business continuity.

Operations

The Operations team is responsible for monitoring the status of IT services during operation, using lightweight ITSM.

Keeping vital services operational in the event of a disaster is critical. The Operations team should involve the reliability engineer and pay attention to two key parameters:

- Recovery point objectives: to which previous point in time is the system recoverable?
- Recovery time objectives: how long does it take to recover the system?

Maintenance

The Service Master and reliability engineer decide together whether to approve maintenance activities. Once maintenance activities are approved, they are added to the product backlog as requests for change (RfC).

Customer service

The Service Master and reliability engineer are responsible together for collecting customer feedback, such as operational issues, including user experience issues, and quality issues. If issues need improvement, these items are added to the product backlog as requests for change (RfC).

End of life

The Service Master has decision power over the end of life of the IT service, including determining conditions for when and how end of life will happen.

Figure 4 Sample configuration of DevOps

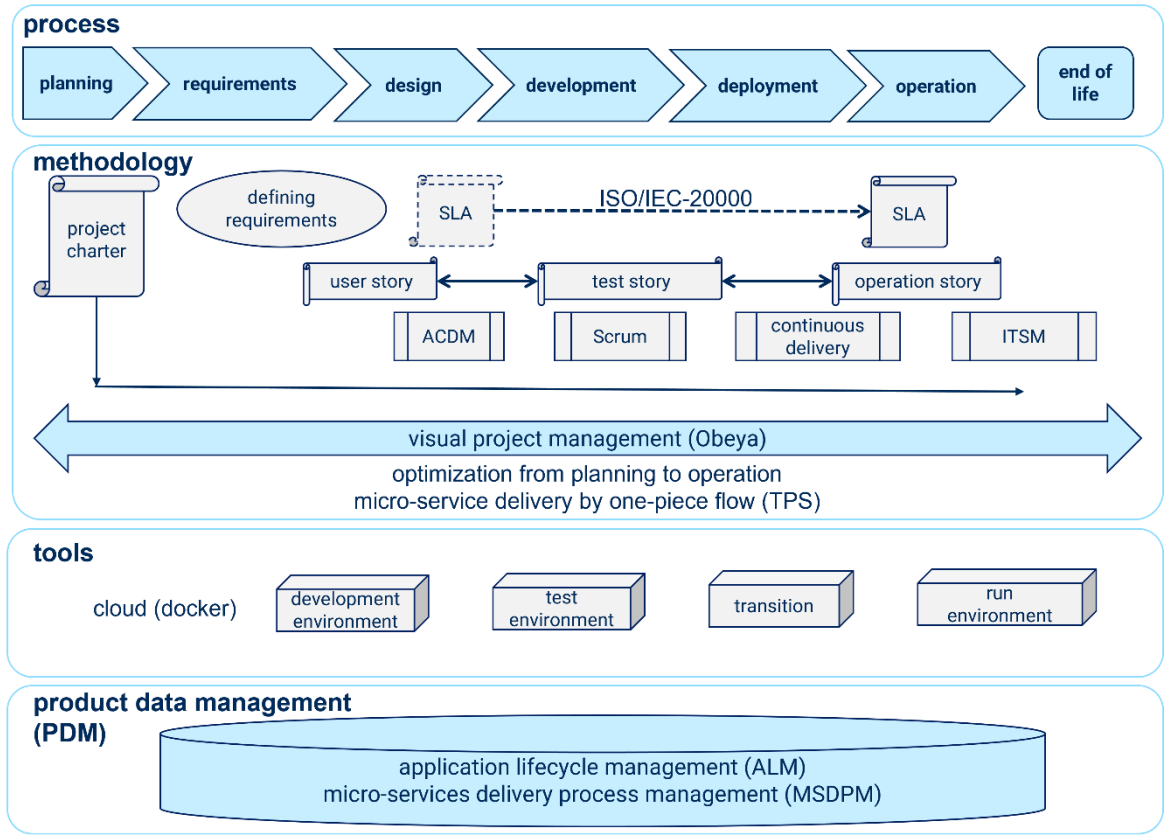
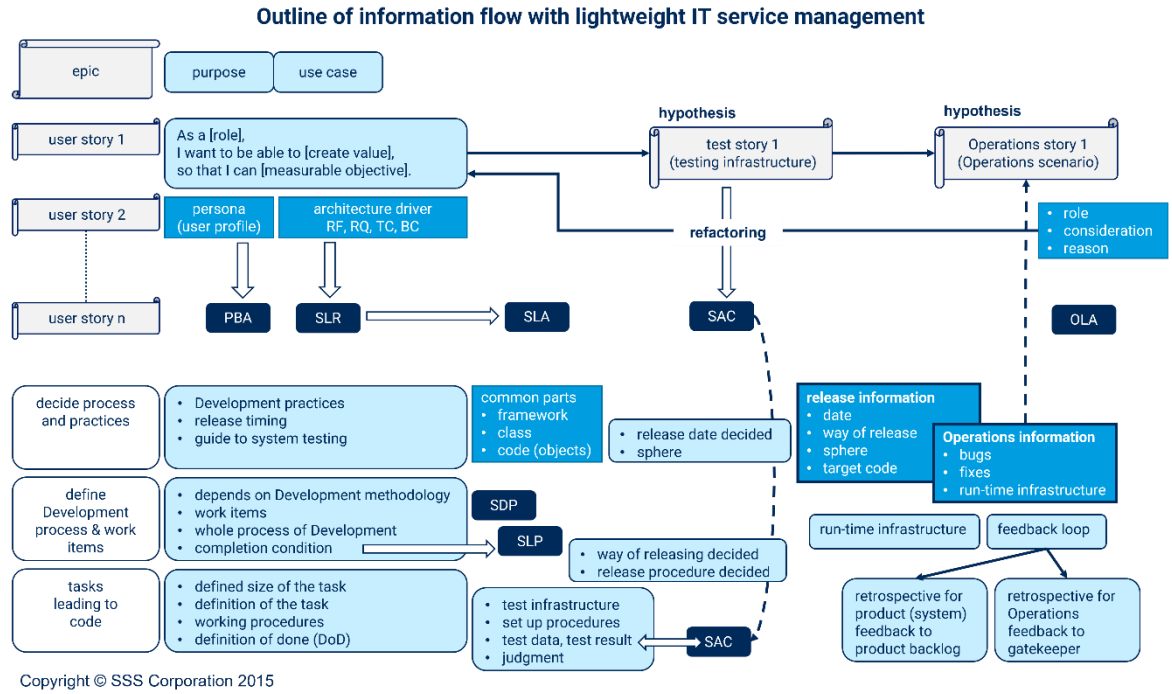


Figure 5 Lightweight IT service management (ITSM)



Abbreviations: RF = required function, RQ = required quality, TC = technology constraint, BC = business constraint, PBA = pattern of business activities, SLR = service level requirements, SLA = service level agreement, SAC = service acceptance criteria, SDP = service design package, SLP = service level package, OLA = operational level agreement,

DevOps implementation

There are three different ways to approach DevOps implementations. Which way is suitable, depends on the business model of the enterprise.

- Toyota way (complex and advanced)
- Collaboration (standard)
- Continuous delivery (basic)

Toyota way

This type of implementation focuses on strategic IT services and gives strategic advantages for the business. It is led by the business owner or Service Master. In a large enterprise, it is preferable to implement a matrix organization and to maintain a close relationship between IT strategy and business strategy.

This way of implementation is most suitable for IT service providers.

Collaboration

This type of implementation focuses on just providing quick, frequent IT services and reliable operation, and is led by the Service Master.

This way of implementation is most suitable for system of engagement (SoE) and system of record (SoR).

Continuous delivery

This type of implementation focuses on quick, frequent releases of software, and is led by the product owner.

This way of implementation is most suitable for digital products vendors.

Conclusion

It is clear that DevOps is a total paradigm shift from most other IT experiences. Therefore, education for staff involved in DevOps is important.

This is the start of your DevOps journey. The certification EXIN DevOps Master™ will help you to understand DevOps better and enables you to gain the benefits that DevOps can bring to your organization.

Glossary

This glossary describes some of the terms as we have used them within the context of this whitepaper.

Kaizen

Continuous improvement means circulating through the PDCA cycle daily or weekly. Root causes of problems or issues can be found by asking "Why?" five times (5 Why).

Problems are defined and supported by data. Does everybody clearly recognize the problem? Set a hypothesis on the problem you found, then think on countermeasures to verify your hypothesis.

Countermeasures must be defined in daily based activities and also need a weekly key performance indicator (KPI) so people can have a sense of accomplishment.

Kaizen (in advance)

When people of the downstream sector recognize a problem caused by the upstream sector, they create possible solutions for removing the problem from the whole process point of view in the most optimal way. Then they propose this solution to the upstream sector. This creates a feedback loop for the problem.

Ji-koutei-kanketsu (JKK)

The concept of JKK is a state of perfection:

- Do not do poor work in your process step.
- Do not accept wrong output from an earlier process step.
- Do not distribute your poor output to the next process step.

The standards for working towards completion by doing it the right way implies that a standard of measurement for the decision to go ahead to the next step must be developed.

TPI Next

TPI Next refers to business-driven test process improvement.

Recommended Literature

For general understanding

Kim, G., Behr, K., & Spafford, G. (2018). *The Phoenix Project: A Novel about IT, DevOps, and Helping Your Business Win.* IT Revolution.

For Process Master, Development team

Davis, J., & Daniels, R. (2016). *Effective DevOps: Building a Culture of Collaboration, Affinity, and Tooling at Scale*. Van Duuren Media.

For Development team, Process Master, Service Master, DevOps Engineer, Reliability Engineer, Gatekeeper

Bass, L., Weber, I. M., & Zhu, L. (2015). *DevOps: A Software Architect's Perspective.* Addison-Wesley.

For DevOps Engineer, Development team, Process Master

Farcic, V. (2016). *The DevOps 2.0 Toolkit: Automating the Continuous Deployment Pipeline with Containerized Microservices* (1st ed.). CreateSpace Independent Publishing Platform.

For Development team, Process Master

Lattanze, A. (2009). *Architecting Software Intensive Systems: A Practitioners Guide. Software Development, Software Engineering, and Project Management.* CRC Press.

For DevOps Engineer, Operations team, Process Master

Duffy, M. (2015). *DevOps Automation Cookbook.* Van Haren Publishing.

For Operations team, Gatekeeper, Process Master

Behr, K., Kim, G., & Spafford, G. (2005). *The Visible Ops Handbook: Implementing ITIL in 4 Practical and Auditable Steps.* Information Technology Process Institute.

For DevOps Engineer, Reliability Engineer, Development team, Process Master

Ewijk, A. van, Linker, B., Oosterwijk, M. van, Visser, B., van Oosterwijk, M., & van Ewijk, A. (2013). *TPI next: business driven test process improvement.* Kleine Uil, Uitgeverij.

For general understanding, Executive level, Manager, Service Master, Process Master

Liker, J. (2020). *The Toyota Way, Second Edition: 14 Management Principles from the World's Greatest Manufacturer* (2nd ed.). McGraw Hill.

For preparation for the EXIN DevOps Foundation certification exam

Skrynnik, O. (2018). *DevOps: A Business Perspective* (1st ed.). Van Haren Publishing.

For preparation for the EXIN DevOps Professional certification exam

Kim, G., Humble, J., Debois, P., Willis, J., & Forsgren, N. (2021). *The DevOps Handbook: How to Create World-Class Agility, Reliability, & Security in Technology Organizations* (2nd ed.). IT Revolution Press.

For preparation for the EXIN DevOps Master™ certification exam

Best, B. D., & Hemmen, L. V. (2022). *DevOps Continuous Assessment: Publication in the Continuous Everything series.* Leonon Media.

Humble, J., & Farley, D. (2010). *Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation.* Pearson Education.

PhD, F. N., Humble, J., & Kim, G. (2018). *Accelerate: The Science of Lean Software and DevOps: Building and Scaling High Performing Technology Organizations* (1st ed.). IT Revolution Press.

Contact EXIN

www.exin.com