EXIN

Success with

Enterprise DevOps

Whitepaper

Koichiro (Luke) Toda
From Strategic Staff Services Corporation
& TPS Certificate Institution
Nobuyuki Mitsui
From Strategic Staff
Services Corporation

EXIN | DevOps

Contents

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

Success with Enterprise DevOps

Enabling IT services to support the business using DevOps.

1.   Introduction

This white paper will use a case study to support the aim of providing information about DevOps. The case study is the use of DevOps framework.

We 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.

DevOps is not just supporting IT. DevOps can also be used to support the business strategy and to improve business processes.

There are many books about DevOps. "The Phoenix Project – IT, DevOps, and Helping your business", is written from the IT manager's perspective. "Continuous Delivery: Reliable Software Release through Build, Test, and Deployment Automation" is written from the development and project manager's perspective. "DevOps A Software Architect's Perspective" is written from the architect's perspective. These are good books to understand what DevOps is.

This white paper is written from a business process perspective, because the intention of DevOps is to establish the software/IT services supply chain to support the business and to manage the whole process to maturity.

A journey to DevOps

The journey started in 2009, after successfully implementing Agile, Scrum and XP methodologies for our client who is providing fancy Web conversion services from PC to mobile devices such as iPhone.

Their Scrum team can develop and release software much more quickly. But the business director confided his concerns that his business speed could not be improved even if the development time had been halved. It appeared that the development process was the bottleneck but on investigation, it was found that the development process was not the bottleneck but 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 stream-lined 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 and collaborate on 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 and improved staff motivation.

This is the true benefit of DevOps.

The DevOps framework should support business outcomes directly, not only collaboration with development and operation for IT services, but because enterprises are using IT services to support and improve their business.

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

2.   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 problem is how the SoR can adapt quickly to changes to the SoE to maintain business continuity. Gartner calls this the Bimodal challenge[2].

The SoR in most enterprises is struggling with the use of legacy applications/systems and can be amended with DevOps by building stream-lined 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 stream-lined processes to enable the business to operate faster and react to changes more quickly. DevOps can also enable maturity by using W.E. Deming's Plan-Do-Check-Act cycle.

[1] See Glossary
[2] http://www.gartner.com/it-glossary/?s=Bimoda

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

3.   What is the goal of DevOps?

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

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 big paradigm shift from software delivery to providing IT services.

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

4.   DevOps Body of Knowledge

When implementing DevOps, here are many sources of knowledge, methodologies, practices, and tools to choose from.

DevOps consists of 3 pillars and a foundation which are described below.

i.   Disciplined Agile

A disciplined Agile development team is key to the success of a DevOps implementation.

Disciplined Agile means:

1. Stabilized Velocity
2. Adaptability for change
3. Always release high quality bug free code.

A more frequent and faster release cycle of IT services to react to business changes depends on the speed of development. The quality of work is the most important item and this can be supported by splitting work into small tasks.

The concept of *Ji-Koutei-Kanketsu (JKK)[3], which means 100% completion of an item, helps keep a high quality of work. The Definition of Done 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 IT service which was done by Toyota's Chief Engineers.

ii.   **Continuous Delivery**

Continuous delivery is the automated implementation of the application build, deploy, test and release processes.

A key focus is on testing such as acceptance testing and performance testing. *TPI NEXT®[4][5] can be useful for improving the maturity of this process.

Every organization will have differences in the implementation of their deployment pipeline depending on their value stream for releasing software.

A Key success factor is to establish only a single deployment pipeline for IT services.

iii.   **IT service management**

As technology is a core component of most business processes, continuous or high availability of IT services is critical to the survival of the business as a whole. This is achieved by introducing risk reduction 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 experience the value that has been promised. Without continuity, the utility (fitness for purpose) of the service cannot be accessed.

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 light-weight 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] See Glossary
[4] TPI NEXT® is a Registered Trade Mark of Sogeti Nederland B.V.
[5] See Glossary

iv. TPS (Lean) concept as foundation

Building a stream-lined supply chain of IT services is difficult because there are many items and it is necessary to change your mindset from the familiar existing development cycle and its methodologies. The concepts of TPS, which includes JIT and automation, can help.
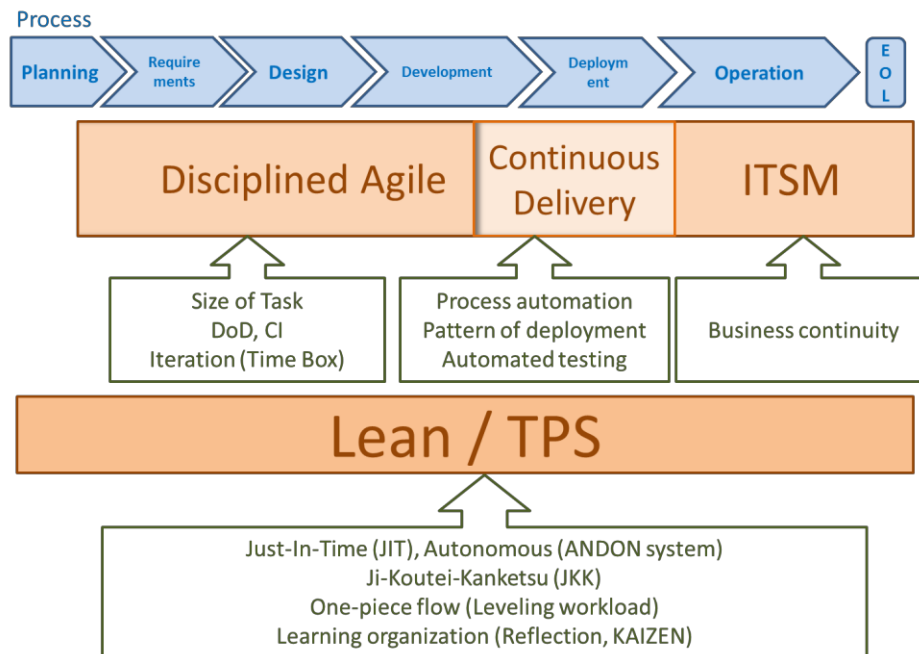
JIT means building up a stream-lined supply chain with one-piece flow. And automation means automating as much as possible and stopping the entire process when a defect occurs.

The process needs to be designed and staff educated for the above two concepts.

The other key issue is the management cycle of Development and Operation. This needs to be changed to work in an Agile way including synchronization between development and operation on a weekly or daily basis.

The diagram below shows the DevOps body of knowledge

Process

| Planning | Require ments | Design | Development | Deployment | Operation | EOL |

| Disciplined Agile | Continuous Delivery | ITSM |

| Size of Task DoD, CI Iteration (Time Box) | Process automation Pattern of deployment Automated testing | Business continuity |

Lean / TPS

Just-In-Time (JIT), Autonomous (ANDON system)
Ji-Koutei-Kanketsu (JKK)
One-piece flow (Leveling workload)
Learning organization (Reflection, KAIZEN)

5.   DevOps Team Roles

It is recommended that a DevOps team is set up in your organization in order to commit to business continuity of the IT service.

It is good to organize small DevOps teams according to Amazon's "Two pizzas rule" i.e. a team small enough to be fed with two pizzas!

The team roles are described below.

Process Master:

Leads the team and facilitates, this role is the same as "Scrum Master" in Scrum. Implements visual control across the entire process and has a strong focus on establishing a stream-lined process with one-piece flow.

Visual control means "Does everybody easily understand the situation by just looking at the boards without explanation?" It does not show the status. It can be expressed the problems occurred or not.

Required experience: Scrum Master, Agile Project Leader.

Service Master:

Has all the responsibility for providing IT services JIT.

This role is like the "Product Owner" in Scrum, which is managing and prioritizing product backlogs and the new additional responsibility of cost planning for the IT service.

Required experience: Scrum Product Owner, Service owner.

DevOps Engineer:

Has a mission to improve and maintain automated process.

The engineer will examine the whole automated process and tools. There are many tools required in the DevOps process.

Required experience: Development and Tools.

Gatekeeper / Release coordinator:

Responsible for monitoring the operational status and progress of the next release of the IT service. Make go/no go decisions about deployment according to criteria including security, compliance, regulatory requirements, maturity of operation team and their process views.

Required experience: IT service management, Operations.

Reliability Engineer (Optional):

Monitor the services in the deployment process and deal with problems with the service during its execution.

Monitor the process status to ensure that the development team are following the rules of CI (Continuous Integration) and CD (Continuous Delivery) strictly.

Monitor and manage the flow of complex build pipeline.

Have a mission to improve the test process.

Required experience: Testing, Tools, Quality assurance.

Development team:

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

Required experience: Development, Agile.

Operation team:

Adopt light-weight ITSM and support the design, implementation, operation, and improvement of these services within the context of an overall strategy.

Utilize "*KAIZEN in Advance"[7] which is a practice of KAIZEN in TPS.

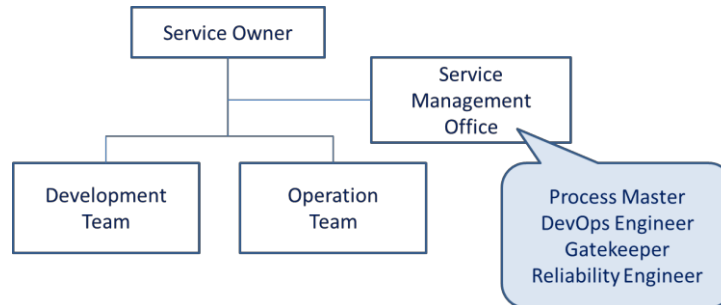Required experience: Operations, KAIZEN.

[7] See Glossary

6. Organization

It is useful to organize the DevOps team in a Service Management Office to support the Service Master.

There are two types of the organizational structure as shown below.

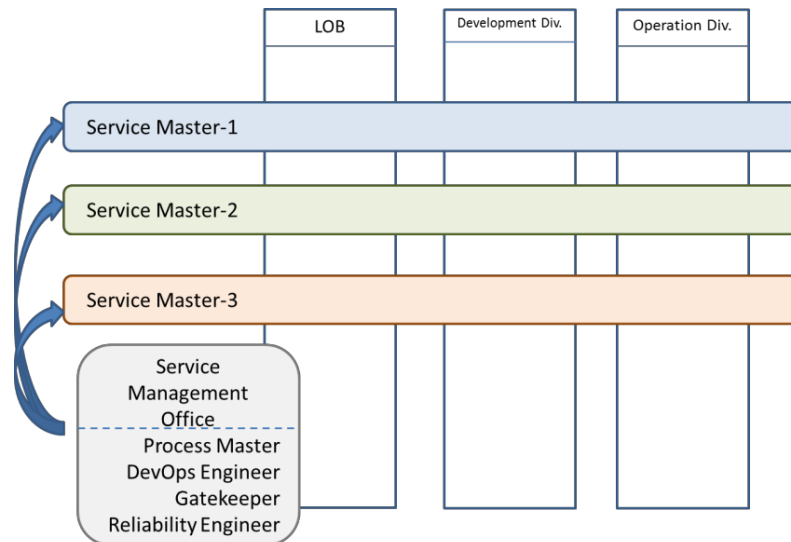i. **Flat organization for small organization.**

This is the basic structure for a small team.

ii. **Matrix organization for large and complex organization.**

Pooling experts and assigning them to each service masters as a team.

The idea for this matrix organization came from the Chief-Engineer in TOYOTA.

7.   DevOps Process

To build up stream-lined processes, JKK is the most effective method to guide the behavior of the DevOps team.

JKK is a way of working with 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.

i.   **Business Strategy and Planning**

IT service has a close relationship with the business strategy and plan.

The service master should attend business planning sessions and make recommendations about how to gain business advantages from IT services.

ii.   **Marketing and sales**

The service master should discuss with marketing how to gain advantages from IT services.

The service master identifies customers of IT services, gathers requirements with business value and agrees a time frame.

iii.   **Administration**

The process master agrees 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 and 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 the cross-functional team works together, the Obeya system enables fast and accurate decision-making, improves communication, maintains alignment, speeds information gathering, and creates an important sense of team integration.

iv.   **Project Planning**

The service master organizes the service management office (SMO) and defines ground rules for the team. The service master creates the vison, goal, and value of the project, and then puts together the DevOps team members.

The run-time infrastructure defined at this stage. A value stream map of the whole process is designed.

v. Requirements and Design

The service master defines product backlogs and priorities.

The DevOps team use the product backlogs to defines stories

- User Story: role, function, business value/reason, and conditions of operation.
- Test Story: acceptance test cases and service acceptance criteria.
- Operation 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 define the transition, test and development infrastructure.

The Development team also creates release and iteration plans.

The gatekeeper studies compliance and regulatory requirements for the IT services.

The reliability engineer defines the testing methodology and test cases.

vi. Development

Scrum is the most applicable methodology 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 agreed according to business need.

From a quality point of view, XP (Extreme Programming) practices such as pair-programming, TDD, Refactoring, and 10 minutes build are effective.

vii. Deployment

After completing continuous integration, the automated process starts for the acceptance test, performance test and deployment.

The DevOps engineer should build the single automated deployment pipeline as a 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.

viii. Operation

The operations team is responsible for monitoring the status of IT services during operation using light weight ITSM process.

Keeping vital services operational in the event of a disaster is critical. The team should involve the reliability engineer and pay attention to two key parameters, Recovery point objective and Recovery time objective.

ix. Maintenance

The service master and reliability engineer decide whether to approve maintenance activities.

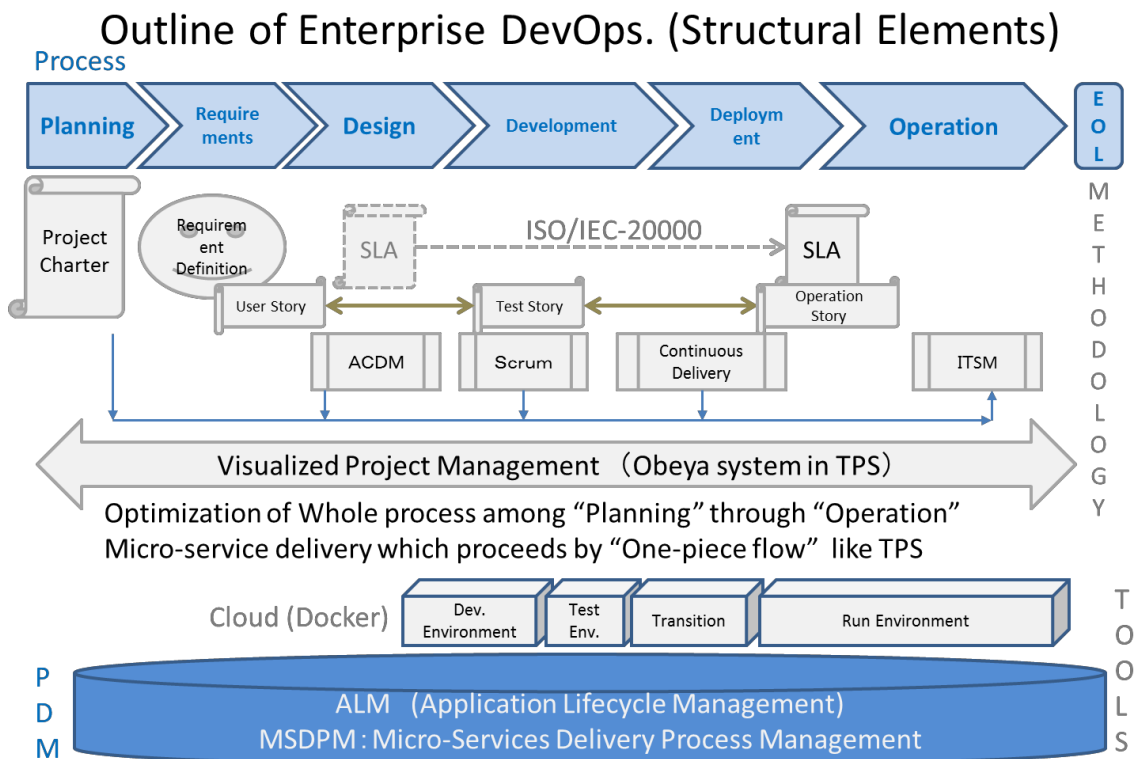If approved, they are added to the product backlog as requests for change.

x. Customer service

The service master and reliability engineer are responsible for collecting customer's feedback such as, operational problems including user experience and quality issues. If approved, these items are added to the product backlog as requests for change.
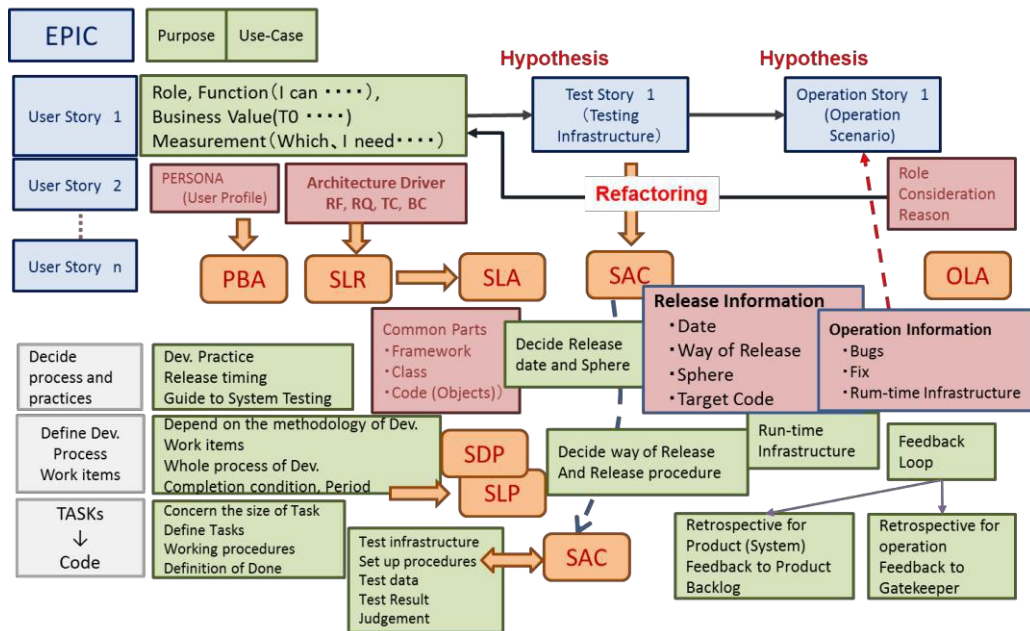
xi. End of life

The service master decides the end of life of the IT service including conditions for when and how this will happen.

The diagram below shows a sample configuration of DevOps.

Outline of Enterprise DevOps. (Structural Elements)

The diagram below shows light-weight ITSM.

Outline of Information flow with light-weight IT Service Management

8. DevOps implementation

There are 3 types of DevOps implementation which are dependent on the business model of the enterprise.

i. **TOYOTA way: (Complex and Advanced)**

This focuses on strategic IT services and gives strategic advantage for the business. It is led by business owner or service master.

It is preferable to implement a matrix organization in a large enterprise and to maintain a close relationship between IT strategy and business strategy. This structure is most suited to IT service providers.

ii. **Collaboration: (Standard)**

This focuses on just providing quick and frequent IT services and reliable operation and is led by the service master.

It is most suited for SoE and SoR.

iii. **Continuous Delivery: (Basic)**

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

It is most suited to Digital Products Vendors.

9. Conclusion

It is now clear that DevOps is a total paradigm shift from most IT experiences.

Education for staff involved in DevOps is therefore important.

This is the start of the journey to your DevOps. "EXIN DevOps Master™" will help you to understand and be

Recommended Literature.

◆ The Phoenix Project
  – A Novel about IT, DevOps, and helping your business.
  Gene Kim, Kevin Behr, George Spafford.
  ISBN9784822285357
  For general understanding

◆ Continuous Delivery
  – Reliable Software Releases through Build, Test, and Deployment automation.
  Jez Humble, David Farley
  ISBN9780321601919
  For DevOps Engineer, Reliability Engineer, Development team, Process Master.

◆ Effective DevOps: Building a Culture of Collaboration, Affinity, and Tooling at Scale
  Jennifer Davis, Katherine Daniels
  ISBN9781491926307
  For: Process Master, Development team

◆ DevOps: A Software Architect's perspective
  (SEI Series in Software Engineering)
  Len Bass, Ingo Weber, Liming Zhu.
  ISBN9780134049847
  For Development team, Process Master, Service Master, DevOps Engineer, Reliability Engineer, Gatekeeper.

◆ The DevOps 2.0 Toolkit:
  - Automating the Continuous Deployment Pipeline with Containerized Microservices.
  Viktor Fracic.
  ISBN9781523917440
  For DevOps Engineer, Development team, Process Master.

- ◆ Architecting Software Intensive Systems (ACDM)
    - A Practitioner's Guide

    Anthony J. Lattanze.

    ISBN9781420045697

    For Development team, Process Master.

- ◆ DevOps Automation Cookbook

    Michael Duffy.

    ISBN9781784392826

    For DevOps Engineer, Operation team, Process Master.

- ◆ The Visible Ops Handbook
    - Implementing ITIL in 4 Practical and Auditable steps

    Kevin Behr, Gene Kim, George Spafford.

    ISBN9780975568613

    For Operation team, Gatekeeper, Process Master.

- ◆ TPI NEXT: Business Driven Test Process Improvement

    Alexander van Ewijk, Bert Linker, Marcel van Oosterwijk, Ben Visser, Gerrit de Vries, Loek Wilhelmus, Tik Marselis.

    ISBN9789072194978

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

- ◆ The TOYOTA Way

    Jeffrey K. Liker

    ISBN9780071392310

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

Glossary

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

KAIZEN: Continuous improvement means circulate the PDCA cycle daily, weekly.

Find a root cause by asking "Why" 5 times.

Problems are defined and supported by data. Does everybody recognize the problems clearly?

Set a hypothesis on the problems you found, then think on counter measure actions to verify your hypothesis.

Counter measure actions must be defined in daily based activities and also need a weekly KPI so people can feel a sense of accomplishment.

KAIZEN in Advance:

When people of the downstream sector recognize some problems caused by upstream sector, they create assumptions for solving the problems as optimum in the whole process point of view. Then they can propose it to the upstream sector. This is a feedback loop of the problems.

Ji-Koutei-Kanketsu (JKK):

Concept of JKK is a State of perfection: Don't do poor work in your process; don't accept wrong output from an earlier process; don't distribute your poor output to the next process.

The standards for working towards completion by doing it the right way means to define a standard of measurement for the decision to go ahead to the next step or not.

TPI NEXT: Business driven test process improvement.