Whitepaper: Light Weight IT Service Management for DevOps

Edition 201704

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

Takashi Maeda
ITIL expert and coach
Contact: maetaka510709@hotmail.co.jp

Content

Light weight IT Service Management for DevOps

1. Introduction

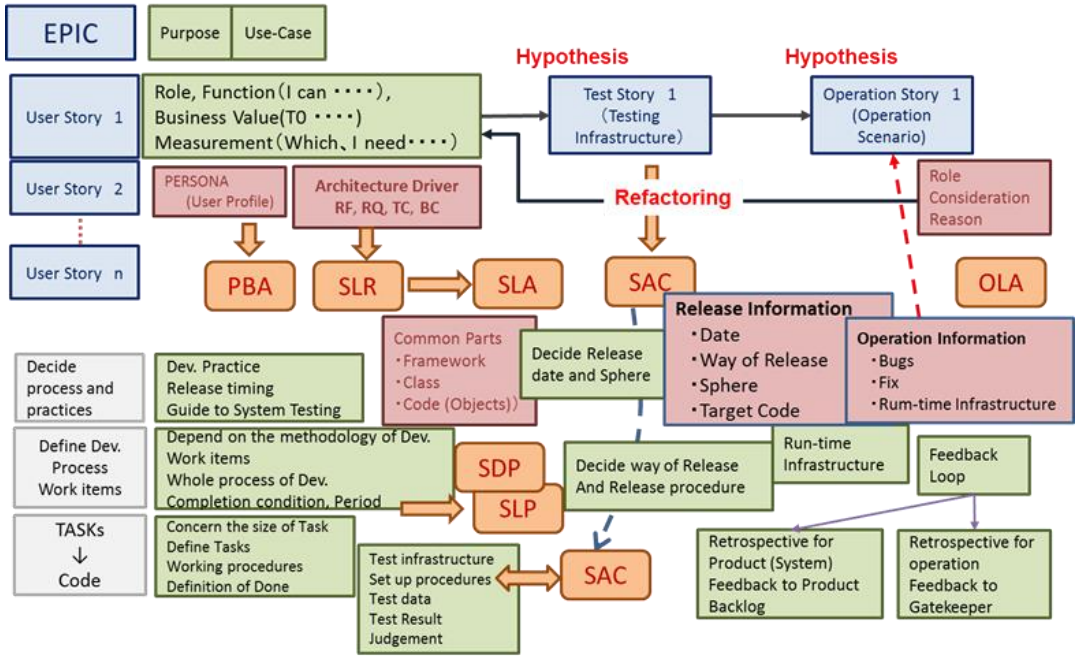
In the field of IT service management, ITIL is established and operated as the way of managing IT system infrastructure aiming for safety and continuity. But as of today, the environment of ITIL is changing by the penetration of Agile and DevOps which require short development cycles and frequent releases by business user's demand. It is difficult to maintain original ITIL management, which is rigid and procedure based, to meet such demand.

We need more light weight and quick IT service management for Agile and DevOps purposes. This is a key issue. We started to examine this issue with an expert in ITIL coaching last year.
The challenge is how to remove inconvenience in order to keep the speed and frequency of Agile. We reached to the conclusion that IT service management should strictly focus on Business continuity.

We reorganized IT service management for Agile development and Lean operations, which means picking up only key information to manage Business continuity elements from IT service management. And we defined these data, which are Patterns of Business Activity (PBA), Service Level Requirements (SLR), Service Level Agreements (SLA), Service Design Package (SDP) Service Level Package (SLP), Service Acceptance Criteria (SAC) and Operational Level Agreements (OLA), generated when and in what process or activities.

The idea is collecting data when they are generated in activities on site and record them. When the data are required from an IT service management point of view, a summary report should be generated and the information can be used. We call that Light weight IT service management. It is not a document, it is just information.

Outline of Information flow with light-weight IT Service Management



The basic thought in Light weight IT service management is generating and collecting Minimum Required Information (MRI) without effort, in order to guarantee Business continuity. The Service owner, Reliability engineer and Operation manager identify each of the data items for Business continuity, because the MRI will be defined by the business environment, the business strategy and the character of the products or the IT service.

This is not changing the way of Agile development. It just adds collecting data for MRI during the work of design and development. Basically, it is not supposed to require additional effort from the team.
Let me explain the process.

2. Planning

The business expresses a service need and the Service owner sets a vision, a goal, a budget, a project scope, and an estimated benefit in the product/project charter. The Service owner and Operations staff discuss and define the Run-time infrastructure for the IT service and the suitable reliability objectives for the IT service. Especially, when implementing the IT service in a Cloud environment, this is an important factor.

We set and configure the Run-time infrastructure at first, then the developer can easily and specifically understand the required performance, the security level, and the reliability of the existing Run-time infrastructure. The developer should develop the right code for the service to work in this environment.

Once the Run-time infrastructure is defined, the service will get a Transition infrastructure, then a Test infrastructure and then a Development infrastructure. On the other hand, when the targeted reliability is defined in this stage, the service will get a clear operation scenario for treating errors and problems in required system's functions such as backup, logging, and duplication. The defined infrastructure in the product/project charter drives the architecture for pre-defined non-functional requirements.

Here is a sample check list of the Service Level Agreement (SLA). These summarized data from the SLA should be included in the product/project charter.

	Classification	Category	Item	Reference Document
a	Authorization	Agreed organization-1	Name of responsible person	
			Signature	
			Position/Role	
			Date of Agreement	
		Agreed organization-2	Name of responsible person	
			Signature	
			Position/Role	
			Date of Agreement	
b	Description of the service	Definition of the service		
		Composition of the service		
		Important business functions		

c	Service scope	Agreed subjects	Targeted System/Service	Mandatory
			Targeted Region/Location	
			Targeted Organization	
			Targeted People	
		Uncovered services		Optional
d	Service hours	Normal service hours		Mandatory
		Exceptions and their conditions		
		Ways of keeping the service alive		
		Service calendar		
		Procedure for changing the service hours		
e	Functionality	Minimum provided services		Mandatory
		Specification of errors and number of the errors allowed to not violate the SLA		
		Level of importance and reporting period/cycle		
f	Service availability	Targeted availability level of the service		Mandatory
		Agreed target figure of availability in normal service hours		
		Period of measuring availability, measuring method of availability		
g	Reliability	Maximum number of allowed interruptions	Monitoring method Recording method	Mandatory
		Mean Time Between Failures (MTBF)		
		Mean Time Between Service Incidents (MTBSI)		
		Definition of Interruption		
h	Service performance	Description of response		Mandatory
		Description of throughput with targeted figure		
		Volume of traffic		
		Throughput		
		Restrictions		
		Reliability		
i	Batch around time	Description of batch around time		Mandatory
		Completed time		
		Description of important outcomes		
		Input time		
		Output time		
		Location		
j	Service continuity	Brief description of continuity plan	Tender Recipient	Mandatory
		Detailed continuity plan and reference to SLA continuity		
		Responsible person for service continuity		
k	Security	Security policy	Tender Recipient	Mandatory
		Responsible person for security		

l	Customer support	Contact method		Mandatory
		Available hours of contact		
		Available hours of support service		
		Target figure of phone call response		
		Target figure of incident response		
		Procedure for extended time frame of support		
m	Escalation	Contact list of the people who are involved		Mandatory
		Description of the escalation process and contact person		
		Definition of complaint and the management procedure of complaints		
n	Change management	Procedure of reference materials and their content		Mandatory
		Definition of the categories for urgency and priorities of change		
o	Responsibility	Description of the responsible person for the service		Mandatory
p	Charging	Description of the way of charging		
		Charging period		
		Reference to the charging policy		
		Procedure for issuing invoices		
		Payment conditions		
		Penalties		
q	Service report/ Review	Contents of the Service report	Frequency Timing Distribution list	Mandatory
		Review Meeting	Frequency Style of the meeting Persons concerned/involved Positions of persons concerned/involved	
r	Glossary	Description of technical terms		Mandatory
s	Revision history	Records of revisions	Details Date of revision Signing person	

Documents of SLA are not created.
These data/records are stored in a file or database as Bill of Services (BOS) when generated.

3. Requirements, Design

At this stage, the User Story including service reliability requirements is used. As you may know the User Story originally includes “Roles (As a role …)”, “Functions (I/We can …)”, and “Business value (In order to …)”. In addition “Condition (Which I need …)” is effective.

After writing the User Story, it needs refactoring to an architecture design point of view. Make sure that architecture drivers such as Required Functions (RF), Required Quality (RQ), Business Restriction (BC), and Technology Restriction (TC) are described.

Once the User Story is fixed, the Operation Story will be created by operations staff. The Operation Story will be presented in such a way that it does not make a difference whether the operator is being trained for the new IT service or not. And it includes any additional or modified configuration of the existing infrastructure as well.

Then the Test Story will be created by a Quality Assurance person or Reliability Engineer consistent with the User Story and the Operation Story. If the Operation Story exceeds the current operation capability, it should be refactored in the User Story. As you already may know, gathering MRIs for IT service management from the User Story, Test Story, and Operation Story is effective.

Especially the User Story will supply beneficial information to IT service management by having a dialogue with the users. So it is good to prepare a check list for the dialogue.

Let me show you an example of the information you can get from

1. **User Story:**
   The Role in the user story will create a description of a user profile (UP).
   The Function in the user story will create information for Service Level Requirements (SLR) and Service Level Agreement (SLA).
   The Role, Function, and Business value in the user story will create information about the Pattern of Business Activity (PBA).
   And the User Story will generate information for the Service Design Package (SDP)/ Service Level Package (SLP) and Service Acceptance Criteria (SAC).
2. **Test Story:**
   The information of the Service Acceptance Criteria (SAC) can go directly from the test scenario and test case to the Test Story.
3. **Operation Story:**
   The information of the Operation Level Agreement (OLA) can go from the environment conditions to the Operation Story with a reference to the Pattern of Business Activity (PBA).

All this information will be available when the work is done and it will be recorded. Furthermore, when Tasks from the User Story are broken down by the agile team, the log of tasks will be useful information for the Service Design Package (SDP). And the Service Acceptance Criteria (SAC) can be verified for keeping quality.

Here is a sample checklist for a Service Design Package (SDP). These data come mainly from the User Story.

	Classification	Category	Item	Reference Document
a	Business matters	Agreed business condition in product/project charter		Mandatory
		Applicability Definition of the service where and how.		
		Contact point of the service	Person in charge of business relationship	
			Contact person for customer	
b	Service design	Requirements for functions of the service (Generated by Epic)	Definition of the service functionality as described in Statement of Requirements (SOR)	Mandatory
		Requirements for service levels (Generate from Epic)	Definition of the service level guaranteed in SLA	Mandatory
		Operational management for the service. (Generate from Epic)	Requirements for the service and its components. Including support, control, operation, measure and report	Mandatory
		Service design and topology-1. (Generate from User Story)	Design for service solution and components	Mandatory
			Definition of the service	Mandatory
			Service model	Mandatory
			Packaging	Mandatory
			Options of the service	Optional
			Service components	Mandatory
			Infrastructure	Mandatory
			Description of business matters/value	
			Description of the service	Mandatory
			Description of components	Mandatory
		Service design and topology-2. (Generate from Release package) Transition and operation of service solution and components	Description of transition	*Optional
			Description of operation	
			Process	
			Procedures	
			Measurements	
			Reports	
			Products for supporting	

			Agreements	
			Suppliers	
c	Assessment	Assessment of organization readiness	Profit for the business	*Optional
			Financial assessment	
			Technical assessment	
			Resource assessment	
			Organizational assessment	
		Assessment of external contacts	Capabilities for contracting with service provider	*Optional
			Capabilities for contracting with supplier	
			Capabilities for contracting with sub-suppliers	
d	Service lifecycle planning	Service program (Generate from Product Backlog) Whole plan or program for covering all steps of the life cycle		*Optional
		Service transition plan (Generate from Release)	Transition strategy	Mandatory
			Way of realization	
			Policy	
			Risk assessment	
			Transition policy	
		Mechanism for building (Generate from Product Backlog)	Building policy	Mandatory
			Conditions of building the service and components with plan	
			Methodology and mechanism specification/ Control / Technology/ Tools/ Platform	
		Mechanism for testing (Generate from Test Story)	Testing policy	Mandatory
			Conditions for test environment and plan	
			Methodology and mechanism Technology / Tools	
		Deployment (Generate from Release)	Deployment policy	Optional
			Release policy	
			Deployment plan	
			Conditions for deployment	

		Acceptance for operation (Generate from Release)	Transition strategy	*Optional
			Way of realization	
			Policy	
			Risk assessment	
			Transition plan	
		Planning for interface and resilience (Generate from Release)	Events	*Optional
			Incidents	
			Problems	
			Errors	
			Issues	
			Disqualification	
		Final service acceptance (Generate from Release)		*Optional
		Criterion of Service acceptance (Generate from Release) Define acceptance criteria in every step of the Service Life cycle for progress of the life cycle process and put to practical use.	All of the related Infrastructure	*Optional
			Term of guarantee	
			Trial period and its criterion	

Note: *mark added to “Optional” in the Reference Document column means it is required when the Release Package defines it mandatory.

Service Design Package (SDP) documents are not created. These data/records are stored in a file or database as Bill of Services (BOS) when generated. From the Application Lifecycle Management perspective, End of Life (EOL) of the IT service can be presumed from the data registered in the BOS, which includes a check list of Service Level Agreements (SLA), Service Design Package (SDP) Service Level Package (SLP), Service Acceptance Criteria (SAC) and Operational Level Agreements (OLA).

4. Development, Deployment

The code developed iteratively in Agile will be available to release. The team should verify the result of testing against the Service Acceptance Criteria (SAC), to define whether the code can be released. The Gatekeeper should create a Release package referring to the Service Design Package (SDP). In an automated deployment pipeline, a check point should be set in each step. The Reliability Engineer or Gatekeeper can examine the state of the IT service and decide whether to move forward, based on the information in the Release package and Service Acceptance Criteria (SAC).

Here is a sample check list of Service Acceptance Criteria (SAC). The data of the Service Acceptance Criteria (SAC) come mainly from the Test Story.

	Classification	Category	Item	Reference Document
a	Date of launch of the service Agreed by all stakeholders			
b	Term of guarantee Agreed by all stakeholders			

c	Criterion of final service acceptance Agreed by all stakeholders			
d	Deployment schedule Documents or information open to the public			Mandatory
e	Service Level Agreement (SLA) / Service Level Requirements (SLR) Reviewed and agreed by all stakeholders			Mandatory
f	Service Input to or updates on the service and check consistency with other components	Service catalog Service portfolio		Mandatory
g	Customers and Stakeholders Distinguished and recorded in Configuration Management System (CMS)			
h	Risk of operation Performed suitable mitigation of risk			Mandatory
i	Correspondence with emergency or extraordinary status Test completed and registered in test schedule of velocity to obstruction	Actions for emergency Actions for fail over		Mandatory
j	Users Defined and approved by all users, appropriate accounts created			Mandatory
k	Load factor and performance Measured all items and put into capacity plan	Live load Performance and capacity		Mandatory

l	Operation Completed and reviewed test documents, then accepted	Operational process Schedule Procedures		
m	Batch operation Completed and reviewed test documents, then accepted	Batch job Printing condition		Mandatory
n	Security Performed appropriately	Security check		Mandatory
o		Security test		
p	Monitoring and measuring Measuring tools and procedures are ready to use			Mandatory
q	Continuous operation	Work related to continuous operation Defined and approved Cost of continuous operation Defined and approved		
r	Cost of operation Incorporated into financial process and cost model			
s	Categories of incidents and problems, and their processes Reviewed or revised known errors and defects of the new service			Mandatory
t	New suppliers Contracted			Mandatory

u	Agreement of support Reviewed and revised by supplier, support team, development team, and other related parties	Service Level Agreement (SLA) Service Level Requirements (SLR) Operational Level Agreements (OLA) Contract		Mandatory
v	Technical support documents Accepted by incident -, problem -, and other IT support teams			Mandatory
w	Request for Change (RFC), Release record Approved and updated			Mandatory
x	Services, Service Level Requirements (SLR), Service Level Agreements (SLA), Operational Level Agreements (OLA), Contracts, Application and components of infrastructure Details recorded in Configuration Management System (CMS)			
y	Software Licenses Verified and assigned			
z	Hardware components Recorded in Configuration Management System (CMS) and stored in fixed Media library			
aa	Release and maintenance Mutually agreed	Plan Release policy Frequency Mechanism		

bb	Users Completed required training and accepted user's documents			
cc	Related documents for acceptable service Documents which are related, internal system and external system, Reliability and Interface, are ready for use, and agreed			Mandatory
dd	Final approval Business Manager approved final acceptance of the new service			Mandatory

Service Acceptance Criteria (SAC) documents are not created.
These data/records are stored in a file or database as Bill of Services (BOS) when generated.

5. Operation

Finally the Gatekeeper decides that the IT service can go into operation, based on the state of sufficiency as registered in the Release Package. After releasing the IT service, the Operation team or Reliability Engineer should feedback the issues or problems to the development team as a Request for Change (RFC). This RFC will be added to the Product Backlog list for the Agile team and the Service owner should manage it with the other backlogs

Contact EXIN

www.exin.com

YOUR ICT COMPETENCE PARTNER