

# Managing IT Service Releases in a Systematic Way: A Case Study Approach

Marko Jäntti  
School of Computing  
University of Eastern Finland  
P.O.B 1627, 70211 Kuopio, Finland  
Email: marko.jantti@uef.fi

Antti Suhonen  
School of Computing  
University of Eastern Finland  
P.O.B 1627, 70211 Kuopio, Finland  
Email: antti.suhonen@uef.fi

Mika Kurenniemi  
School of Computing  
University of Eastern Finland  
P.O.B 1627, 70211 Kuopio, Finland  
Email: mika.kurenniemi@uef.fi

**Abstract**—Release management is responsible for planning, scheduling and controlling the deployment of releases to test and live environments. In many IT service provider organizations, the IT service release management is a very actual improvement target. Process frameworks, such as IT Infrastructure Library (ITIL), are often used as a basis of the process improvement. The research problem of this study is: How IT service releases can be managed in a systematic way? The main contribution of this paper is to present results of a case study with a Nordic IT service provider organization.

**Keywords**—IT service; release management; process

## I. INTRODUCTION

A *release* is a collection of hardware, software, documentation, processes or other components required to implement one or more approved changes to IT services [1]. Releases can be categorized into major releases, minor releases and patches. Release management activities should be conducted within the release management process that is coordinated by a release manager. A systematic approach for release management provides the following business benefits:

- delivering changes faster and at optimum cost and minimized risk [1]
- fewer releases to be rolled out to customers [2]
- releases are promoted successfully, are stable and meet expectations [3]
- releases are delivered according to agreed release policy and planned release cycles

There are three key challenges related to release management improvement from IT service management perspective. First, release management is often not seen as a process but is conducted in the form of separate activities, such as installations and packaging. This causes challenges for people who would like to improve the process because they cannot just go to employees and ask how they perform release management because employees do not know what is included in managing releases. Second, IT service organizations often lack the consistent understanding what is a release and how it is related to projects, service requests and change requests. Lack of understanding may lead to the following types of questions:

- Does a release cover installations required by a service request handling?

- Can we consider the project outcome of a deployment project as a release?
- Should every change implementation be treated as a release?

Third, a weak release management process typically leads to a fact that information on installations or releases is stored somewhere else than release records such as in change management.

Because ITIL is a best practice framework and not a standard, IT companies may aim at certifying their service management based on ISO/IEC 20000 standard family. The most popular parts of the standard family are ISO/IEC 20000-1:2010 Part 1: Service management system requirements [4] and ISO/IEC 20000-2:2011 Part 2: Guidance on the application of service management systems [5]. ISO/IEC TS 15504-8:2012 process assessment model [6] can be used to measure or improve the service management process capability. 15504-8:2012 provides the following base practices for release management [6]:

- Establish requirements for releases.
- Plan releases of services or service components.
- Design releases.
- Test releases.
- Deploy releases.
- Assure integrity of hardware, software, and other service components during deployment of the release.
- Reverse or remedy unsuccessful releases.
- Communicate release information to interested parties.

Much has been written about service management from service operation perspective. However, surprisingly few of studies have dealt with release management practices in IT service provider companies There are some studies that have focused on software release management such as the study of van Der Hoek and Wolf [7] that addresses requirements for release management: *...The release process should involve minimal effort on the part of the developer...The scope of a release should be controllable...* Jansen and Bringkemper [8] discuss common misconceptions about product software release management.

Jokela and Jäntti [9] have identified challenges in release management process from product portfolio management perspective. They report that challenges were related to unclear release and deployment management and/or product portfolio management process roles, lack of process for product portfolio release and deployment management, lack of communication between product managers and lack of resources and time for product portfolio integration, testing and reviewing. There are studies that use the term patch management instead of release management, such as the study of Liu et al. [10] which presents methods for effective patch management. Jäntti and Sihvonen [11] have examined the patch management within release management. They observed that challenges exist especially in release management concepts and classifications. Patch management can be seen as a subprocess of release management.

A. Our Contribution

The main contribution of this study is

- to show how release management activities are performed in a Finnish IT service provider organization,
- to provide lessons learnt from release management process improvement.

The results of this study might be useful for release and deployment managers, installation team managers and other IT service management process managers. The remainder of the paper is organized as follows. In Section II, the research methods of this study are described. In Section III, case study results are presented. Section IV is the analysis of findings. The discussion and the conclusions are given in Section V.

II. RESEARCH PROBLEM & METHODOLOGY

The case study was conducted during KISMET (Keys to IT Service Management and Effective Transition of Services) research project in May - June 2013. The research problem of this study is: How IT service releases can be managed in a systematic way? The research problem was divided into the following research questions:

- Which factors trigger the release management?
- How release management activities are performed in the case organization?
- What types of releases exist in the organization?
- How release management should be implemented with an IT Service Management tool?

A case study research can be defined as "a research strategy focusing on understanding the dynamics present within single settings"[12]. Runeson and Höst [13] state that studies can be categorized into four types: 1) exploratory studies that focus on finding out what is happening, seeking new insights and generating ideas and hypotheses for new research, 2) descriptive studies that focus on portraying a situation or phenomenon, 3) explanatory studies focusing on seeking an explanation of a situation or a problem and 4) improving studies that aim to improve a certain aspect of the studied phenomenon. Our study could be classified as an exploratory

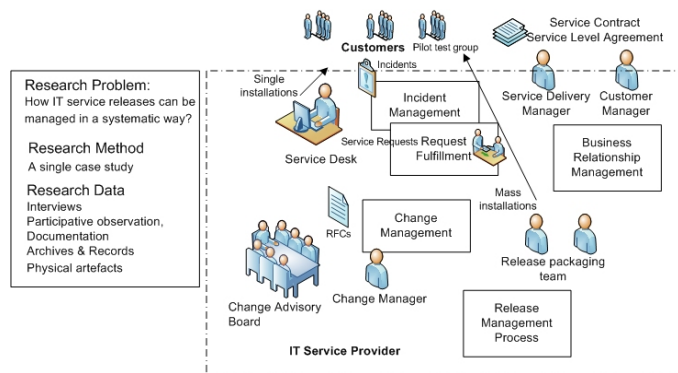


Fig. 1. The context of the case study

and improving case study. A case study research method with a single case was used to answer the research problem. Figure 1 shows the context of the case study.

A. The Case Organization and Data Collection Methods

Our case organization Alpha is a Nordic IT service provider company that provides IT outsourcing services and IT consulting services in Finland, Sweden, Norway and Denmark. Alpha has around 800 employees. The case study focused on exploring release management activities especially in workstation management service area. The company uses IT Infrastructure Library -based service management processes in incident management, problem management and change management. Release management was a natural choice for the improvement target because it is responsible for implementing changes.

The case study started with a kick-off meeting in May 2013 where improvement goals were discussed. The main objectives of the improvement pilot were to explore how release management activities can be performed in practice, how release management could be implemented to the ITSM system and describe the process from a change to a release that is delivered to a customer.

Yin's [14] data collection principles were used to increase the quality of the data collection: Data was collected by three researchers using multiple sources of evidence in Alpha's facilities. A case study datastore was established and maintained during the study. Because NDAs were signed between a research team and the case organization, only three researchers were able to investigate the case study material. A chain of evidence was established by recording data sources (persons and their roles, date of data collection, document name) and linking findings to data sources. The following sources of evidence were used:

- Documentation (change plan, change task models, a list of standard changes, application package order form, image order form, workstation management service descriptions).
- Archives (Change request records, service request records)
- Interviews/discussions (change process owner, 2 change managers, CSI manager, release packaging team member)

- Participative observation (release management meetings)
- Direct observation (a Change Advisory Board meeting)
- Physical artefacts (Installation manager tool demonstration, access to development environment of the ITSM tool)

### B. Data Analysis Method

The case study data was collected and analyzed by three researchers using a within case analysis technique [12]. Research findings were validated in two meetings with the representative of the case organization. The within-case analysis resulted in a case study writeup that was delivered to the case organization. The document summarized the case study findings and improvement actions.

### III. IT SERVICE RELEASE MANAGEMENT: CASE STUDY FINDINGS

Next, a summary of the case study results is presented. In this paper, we focus on release management activities although the case provided a lot of findings related to the change management process.

#### A. Which factors trigger the release management?

We consider a Change Advisory Board and change managers as primary triggers for release management. The Change Advisory Board is a group of people that advises the change manager in the assessment, prioritisation and scheduling of changes [1]. Regarding authorization of changes we observed that change managers bring all the normal changes to CAB. In ITIL it is possible that a change manager may authorize the change without CAB meetings.

New standard changes are brought to CAB for preauthorization like in ITIL. After that they are typically handled in a service request fulfillment process. We observed that some installations are triggered by application package orders (Order form for application packages). A customer manager usually fills the form together with a customer and delivers the form to the service desk that submits the form to the packaging teams's queue. The order form for application packages defines the following details of the application to be packaged:

- Application name
- Number of users
- Application super user
- Application provider
- Description of application
- Storage for application media
- Installation code
- Language version
- Release method
- Operation system requirements

- Details of application package testing
- Target of release
- Change plan

Although these installations look like releases, it may be wise to exclude them from release management scope and record them as a part of request fulfillment process. However, normal changes that are processed by CAB could be scheduled and linked to a release. The case organization also seemed to lack the major change concept. We interpreted that a change with a major impact is equivalent to a major change.

There is a statement in ISO/IEC 20000-1 standard [4] that *requests for change classified as having the potential to have a major impact on the services or the customer shall be managed using the design and transition of new or changed services process*. A major change may occur in case of a new customer, a new customer for an existing service or a change that affects a certain number of users. Additionally, we may interpret that an emergency change is a change that receives the highest urgency level. We found an emergency change procedure in change management process description.

#### B. How release management activities are performed in the case organization?

The following observations were captured from the release management interviews with the case organization's employees:

- Change managers shall prepare the changes for the Change Advisory Board, a change manager can also reject a change.
- The biggest challenge is that there is no owner information regarding the computer the release should be delivered to.
- Customer might buy computers where we cannot put any images on.
- The request for a new release package may come from a customer through the service desk (application package form).
- If the form is poorly filled, a packaging team member shall retrieve the information.
- At the moment, Alpha does not have a change calendar.
- There are two tools used for installing software packages. The new one enables centralized installations, the old one requires establishment of customer site.
- Regarding the reports, customers are mainly interested in software usage level and application inventory (how many computers have a specific application version).
- Change and release schedules are agreed with customers by a customer manager / project manager / service delivery manager.
- Alpha has a small packaging team, thus a lot of issues shall be solved by discussions.

- An unsuccessful release is a release that fails to be installed to the computer. In case of a more complicated product, a user may inform the service desk that the application does not work.
- Major release updates shall be tested with all applications that need the update
- Alpha does not have a release note but information is stored in configuration management tool.

The following lifecycle for an installation was defined by the research team:

- Alpha’s customer indicates the need for software distribution to a customer manager, or directly contacts the Service Desk (SD).
- Alpha’s customer manager fills the billing information and submits the form to SD or SD fills out the order form based on the information given by a customer.
- SD controls the order of the packaging team queue in the ITSM tool (if the form has information gaps, the packaging group specialist calls for more information). packaging group builds a software package and tests it before using it.
- In order to deploy a release, the packaging group distributes the software package initially only specified customer (test) persons.
- If the distribution goes successfully to customers and they do not report any problems, then after a pre-defined time period distributions shall be done for all computers.

C. What types of releases exist in the organization?

Two different tools were used in the case organization to install software packages to customers. The research team participated in the demonstration of the new installation tool and identified the following types of releases:

- Audit (for example, google chrome updates)
- Configuration (java runtime environment, disable /enable java update)
- Critical updates (windows critical updates)
- Deploy (Windows program removal tool)
- Feature Pack (Windows patches, platform update)
- Hotfix (update for .Net framework)
- Microsoft unsupported (no more official support available for these releases)
- Rollup (collection of product updates)
- Security Advisory (single security updates)
- Security Hotfix (vulnerabilities in MS application)
- Security Update (application security updates)
- Service Pack (includes updates)

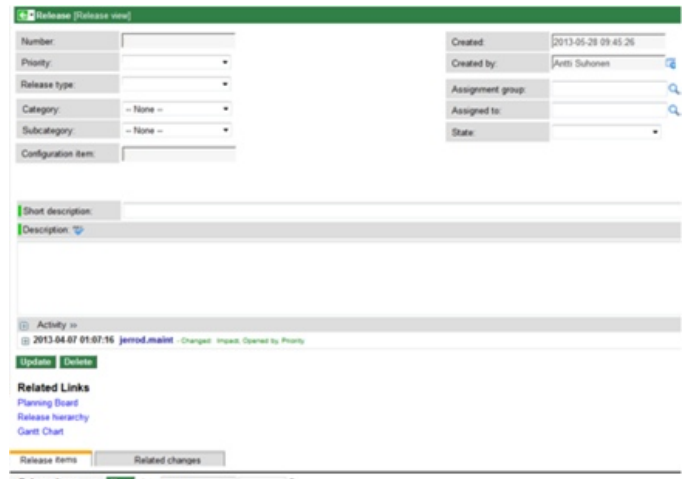


Fig. 2. The draft version of the release record

D. How release management should be implemented with an IT Service Management tool?

The organization had recently changed their ITSM tool and had implemented incident management, service request management and change management to the new tool. However, the release management module had not been in use. One of the research team’s tasks was to explore how release management could be implemented with a tool. Researchers spent a lot of time to look at change management module and its operational behavior.

Main observations from the tool side were the release module requires, for example, a button that enables creating a release from a change request, a user interface element that shows which change requests are related to a particular release, a release type field, release tasks that follow the release management process phases (for example, in planning, in testing) and finally hiding the Features. A Feature was a tool-related concept initially visible in release management user interface. A consultant from the tool provider side recommended hiding the concept to make the process simpler.

At the beginning, the difference between release items and release tasks was a little bit unclear to the research team. We interpreted that release items referred to the structure of releases and release tasks to the release management activities. Figure 2 shows the draft version of the release record.

At the end of the improvement pilot, the research team had a meeting with the ITSM tool development team. The result of the discussion was that most of the improvement ideas that the research team had suggested were implementable. The tool development team advised researchers to create RFCs to the Change Advisory Board of the ITSM tool.

IV. ANALYSIS

A within-case analysis technique was used in this study. This study showed that release management process improvement in IT service provider context is far away from a simple case. Release management process improvement is typically based on best practices of ITIL. It seems that the release management process is easier to be adopted by software

providers than IT service providers. The following lessons learnt were derived from the case study.

**Lesson 1: Strong change management affects the role of release management.** When a change management process is deployed before release management, this may lead to a situation where change management may become relatively stronger process area than release management. There is a risk that keeping release management as a subpart of change management process, decreases the visibility of release management aspects. In our case, the most release-related information was stored in change tasks because there was no release record available.

**Lesson 2: Transition of new or changed services is a complicated area.** Transition of new or changed services is a process area in ISO/IEC 20000 standard [5]. This process area is related to release management in the following way: *...The transition of services should include the build, test and acceptance of the new or changed services followed by making the new or changed services operational through the release and deployment management process....* We observed that both design and transition of new or changed services would have required clarification. We aim to clarify this issue by stating that building of a new service can be seen as a major change. The implementation of a major change should be carried out as a project the outcomes of which form a release.

**Lesson 3: Establish a release record.** In early phase, we observed that there was no release record or release note practice in use. The release record could be visible to customers and show for example which incidents have been resolved by the particular release. In order to get change management and release management to support better ITSM best practices, a change record should have a field that allows the creation of a release. This Release button should be set visible not until the CAB has authorized the change. Basically, the button works in a same way than creating a problem record based on an incident.

The release type field may include four simple categories as a starting point: Major Release, Minor Release, Patch, and Fix. The release record should also guide the user to implement release according to predefined release tasks. The ITSM tool can be configured in such a way that a task needs to be completed before a new task can begin. To create traceability between installation tool and ITSM tool, one could add an action id of the installation to the release record of an ITSM tool.

**Lesson 4: Implement a release schedule.** One of our findings was that there was no clear release schedule that would show the frequency of releases. The research team recommended implementing a release schedule and communicating it to customers and staff such as service desk workers. There was evidence that some service areas in the case organization used maintenance windows that were communicated to customers.

**Lesson 5: Define an emergency release procedure.** The ISO/IEC 20000 standard requires that there is a documented procedure for managing emergency releases. We defined a very abstract level procedure:

- The need for emergency release is identified

- Every employee can make a decision on building an emergency releases
- Emergency releases shall be tested in a very light mode
- Emergency release shall be deployed to live environment
- Emergency release information shall be recorded in the ITSM system
- Emergency change shall be approved afterwards

**Lesson 6: Assign a release manager role.** According to our findings the organization does not have a release manager. Process managers have important roles both in ensuring that the process runs smoothly and monitoring and measuring the process. Sharifi et al. [15] have explored why ITIL implementations fail. One of the factors was not assigning process owners. The case organization should clarify who is responsible for the whole release management process. This role should be responsible for [2]: producing management reports, creating and maintaining release and deployment policies, providing reports on the progress of releases and ensuring that release management follows the organization's procedures and policies. A smaller organization might combine the role with a change or configuration manager role.

The above mentioned list is based on our findings from the case organization Alpha and lessons learnt are not presented in a priority order. This was the second case study on release management improvement with the case organization. In our first case study [9], the case organization had product-oriented business focus compared to Alpha that is a service provider. However, we observed same type of challenges, such as difficulties in defining a release policy. The main difference we observed was that in the product-oriented release management releases are defined by product features while in IT service release management releases are defined by requests of change.

## V. CONCLUSION

The research problem of this study was: How IT service releases can be managed in a systematic way? The main contribution of this study was to explore release management activities in a Nordic IT service provider organization. The key improvement ideas we identified were related to classification of releases, understanding the difference between a release and a change request, release management coordination by a release manager, and implementing a release record to the ITSM tool.

This case study included certain limitations. First, data were collected by using qualitative case study research methods from one service area. Quantitative case study methods could have been applied to examine the number of failed changes and releases. Second, we used a convenience sampling as a case selection criteria. The research team had easier access to the case organization because they were an industrial partner of the research team. Further research could explore the release management interfaces with other service management processes such as configuration and change management.

## ACKNOWLEDGMENT

This paper is based on research in KISMET project funded by the National Technology Agency TEKES (no. 70035/10), European Regional Development Fund (ERDF), and industrial partners. Special thanks to the case organization's staff for providing feedback, material and facilities to the research team.

## REFERENCES

- [1] Office of Government Commerce(c), *ITIL Service Transition*. The Stationary Office, UK, 2007.
- [2] Office of Government Commerce, *ITIL Service Delivery*. The Stationary Office, UK, 2002.
- [3] COBIT 5, *Control Objectives for Information and related Technology: COBIT 5: Enabling Processes*. ISACA, 2012.
- [4] ISO/IEC 20000:1, *Part 1: Service management system requirements*. ISO/IEC JTC 1 Secretariat, 2010.
- [5] ISO/IEC 20000:2, *Part 2: Guidance on the application of service management systems*. ISO/IEC JTC 1 Secretariat, 2011.
- [6] ISO/IEC TS 15504-8:2012, *Information technology - Process assessment -Part 8: An exemplar process assessment model for IT service management*. ISO/IEC TC JTC1/SC7 Secretariat, 2012.
- [7] A. van der Hoek and A. L. Wolf, "Software release management for component-based software," *Softw. Pract. Exper.*, vol. 33, no. 1, pp. 77–98, 2003.
- [8] S. Jansen and S. Brinkkemper, "Ten misconceptions about product software release management explained using update cost/value functions," in *Proceedings of the International Workshop on Software Product Management*. Washington, DC, USA: IEEE Computer Society, 2006, pp. 44–50.
- [9] K. Jokela and M. Jäntti, "Challenges and problems in product portfolio release and deployment management," in *Proceedings of the 9th International Conference on Service Systems and Service Management (ICSSSM12)*. Shanghai, China: IEEE, 2012.
- [10] S. Liu, R. Kuhn, and H. Rossman, "Surviving insecure it: Effective patch management," *IT Professional*, vol. 11, no. 2, pp. 49–51, march-april 2009.
- [11] H.-M. Sihvonen and M. Jantti, "Improving release and patch management processes: An empirical case study on process challenges," *Proceedings of the International Conference on Software Engineering Advances (ICSEA 2009)*, vol. 0, pp. 232–237, 2010.
- [12] K. Eisenhardt, "Building theories from case study research," *Academy of Management Review*, vol. 14, pp. 532–550, 1989.
- [13] P. Runeson and M. Höst, "Guidelines for conducting and reporting case study research in software engineering," *Empirical Software Engineering*, no. 14, pp. 131–164, 2009.
- [14] R. Yin, *Case Study Research: Design and Methods*. Beverly Hills, CA: Sage Publishing, 1994.
- [15] M. Sharifi, M. Ayat, A. A. Rahman, and S. Sahibudin, "Lessons learned in ITIL implementation failure," in *Information Technology, 2008. ITSIM 2008. International Symposium*, vol. 1, aug. 2008, pp. 1–4.