Release and Deployment Management using ITIL

G.Rasa[1], S.Jagadheesh Kumar[2], Dr.R.S.D.Wahida Banu[3]

GJCST Classification D.2.9

Abstract **- Release management is the process of determining, acquiring, releasing and deploying changes into an Information Technology (IT) environment. The Service Delivery management strategy provides support to the Service Support process and ensures that the required services are delivered to the customer on time. Information Technology Infrastructure Library (ITIL) is a framework available in the form of documentation that defines the best-practices and approaches to manage IT services. This paper describes how to release a change into the IT environment using ITIL.**

Keywords- ITIL, Change Management, Configuration Management, Release Management, Deployment Management.

I. INTRODUCTION TO ITIL

ITIL is the most widely adopted approach for IT Service Management in the world. It provides a practical framework for identifying, planning, delivering and supporting IT services to the business This is a compendium of best practices from many companies in many industries. It represents the best thinking of thousands of people about how IT should be run, what the impact IT can have on the business it supports, and how to gain the most value from your IT investments. It provides guidance to organizations on how to use IT as a tool to facilitate business change, transformation and growth. One of the stated goals of the ITIL is to help decision makers make better decisions by ensuring the adequate IT information is available to support those decisions.

Adopting ITIL can offer users a huge range of benefits that include:

- improved IT services
- reduced costs
- improved customer satisfaction through a more professional approach to service delivery
- improved productivity
- improved use of skills and experience
- improved delivery of third party service.

ITIL is the product of the Office Government of Commerce (OGC), United State. ITIL originally emerged in the mid to late 1980s. The CCTA (Central Computer and Telecoms Agency) was a major UK government department, with an IT budget of around £8 billion. ITIL fulfill the ISO standard: ISO/IEC 20000. There are three versions of ITIL available which are 1.0, 2.0 and 3.0 . Among them, version 2 and 3 are the key versions. The difference between the two versions is a few changes in service life cycle structure [1]. Fundamentally, ITIL is exactly what its name implies "a collection of books" The common theme of the library is that all of the books provide guidelines that can help organizations implement the best practices that have been learned the hard way by the pioneering few. The library continue to grow as more successful techniques are documented and guidelines established for what can make others successful.

1) *ITIL version 2.0*

In 2000/2001, to make ITIL more accessible, ITIL v2 consolidated the publications into 8 logical "sets" that grouped related process-guidelines to match different aspects of IT management, applications, and services. However, the main focus was known as the Service Management sets (Service Support and Service Delivery) which were by far the most widely used, circulated, and understood of ITIL v2 publications.

- In April 2001 the CCTA was merged into the Office of Government Commerce (OGC), an office of the UK Treasury.[2]
- In 2006, the ITIL v2 glossary was published.
- In May 2007, this organization issued the version 3 of ITIL (also known as the ITIL Refresh Project) consisting of 26 processes and functions, now grouped under only 5 volumes, arranged around the concept of Service lifecycle structure.
- In 2009, the OGC officially announced that ITIL v2 would be withdrawn and launched a major consultation as per how to proceed.[3]

a. Service Support
b. Service Delivery
c. ICT Infrastructure Management
d. Planning to Implement Service Management
e. Application Management,
f. The Business Perspective Management
g. Security Management[4]

2) *ITIL Version 3.0*

The following are the five guides comprise the ITIL v3, published in May 2007:

a. ITIL Service Strategy
b. ITIL Service Design
c. ITIL Service Transition
d. ITIL Service Operation
e. ITIL Continual Service Improvement

About[1] - Research Scholar, Dravidian University, Kuppam, AP, India.
About[2]- Project Manager, IBM India Pvt. Ltd, Bangalore, India. Research Scholar, Dravidian University, Kuppam,KA,India.
About[3]- Professor & Head, Department of Electronic and Communication Engineering, Government College of Engineering, Salem,TN, India.

The five core guides map the entire ITIL Service Lifecycle, beginning with the identification of customer needs and drivers of IT requirements, through to the design and implementation of the service into operation and finally, on to the monitoring and improvement phase of the service.

3) *ITIL Service Strategy*

As the center and origin point of the ITIL Service Lifecycle, the ITIL Service Strategy volume[5] provides guidance on clarification and prioritization of service-provider investments in services. More generally, Service Strategy focuses on helping IT organizations improve and develop over the long term. In both cases, Service Strategy relies largely upon a market-driven approach. Key topics covered include service value definition, business-case development, service assets, market analysis, and service provider types.
List of covered processes:

a. Financial management
b. Demand Management
c. Service Portfolio Management(Available Version 3.0 only)

4) *ITIL Service Design*

The ITIL Service Design volume[6] provides good-practice guidance on the design of IT services, processes, and other aspects of the service management effort. Significantly, design within ITIL is understood to encompass all elements relevant to technology service delivery, rather than focusing solely on design of the technology itself. As such, Service Design addresses how a planned service solution interacts with the larger business and technical environments, service management systems required to support the service, processes which interact with the service, technology, and architecture required to support the service, and the supply chain required to support the planned service. Within ITIL v2, design work for an IT service is aggregated into a single Service Design Package (SDP). Service Design Packages, along with other information about services, are managed within the service catalogs.
List of covered processes:

a. Service Catalogue Management
b. Service Level Management
c. Risk Management
d. Capacity Management
e. Availability Management
f. IT Service Continuity Management
g. Information Security Management
h. Compliance Management
i. IT Architecture Management
j. Supplier Management

5) *ITIL Service Transition*

Service transition, as described by the ITIL Service Transition volume,[7] relates to the delivery of services required by a business into live/operational use, and often encompasses the "project" side of IT rather than "BAU" (Business as usual). This area also covers topics such as managing changes to the "BAU" environment.
List of processes:

a. Service Asset and Configuration Management
b. Service Validation and Testing
c. Evaluation
d. Release Management
e. Change Management
f. Knowledge Management ITIL Service Operation

6) *Service Operation*

Best practice for achieving the delivery of agreed levels of services both to end-users and the customers Service operation, as described in the ITIL Service Operation volume,[8] is the part of the lifecycle where the services and value is actually directly delivered. Also the monitoring of problems and balance between service reliability and cost etc. are considered. The functions include technical management, application management, operations management and Service Desk as well as, responsibilities for staff engaging in Service Operation.
List of processes:

a. Event Management
b. Incident Management
c. Problem Management
d. Request Fulfillment
e. Access Management

7) *ITIL Continual Service Improvement*

The continual service improvements have seven methods to improve the service such as

a. Define what you should measure?
b. Define what you can measure?
c. Gather the data
d. Process the data
e. Analyze the data
f. Presenting and using the data
g. Implement the corrective action

II. RELEASE AND DEPLOYMENT MANAGEMENT

1) *Terminology:*

Release: A release is a collection of authorized changes to an IT service. i.e., A collection of hardware, software, documentation, processes or other components required to implement one or more approved changes to IT services. The contents of each release are managed, tested and deployed as a single entity.
Release Unit: A ‚release unit' describes the portion of a service or IT infrastructure that is normally released together according to the organization's release policy.
Release Package: A release package may be a single release unit or a structured set of release units, including the associated user or support documentation that is required. Like the definition of release units, factors such as the

modularity of components, the amount of change occurring and resources required will be considered when formulating a complete Release Package[13]

Change Management: Change management processes are used to deliver a finalized and tested change into a pre-production environment along with a set of tools and/or procedures for migrating the change into the live production environment [9].

Build Management: The software, hardware and documentation that comprise a release unit should be assembled in a controlled manner to ensure a repeatable process. This should include automation where possible for its compilation and distribution, which for large organizations can significantly reduce the Total Cost of Ownership for the services involved.

Release Management: Release Management is an important key technology for distributing the project/product to the customer.[12]

Deployment: The activity responsible for movement of approved releases of hardware, software, documentation, process etc. to test and production environments.

2) *Release and Deployment Management*

The process responsible for planning, scheduling and controlling the movement of releases to test, pre-production and production environments. The primary objective is to ensure the integrity of the production environment.In conjunction with the use of Change Management, Release and Deployment will enhance an organization's capabilities to develop, compile, reuse, distribute and rollback releases in accordance with defined policies that improve efficiency and reduce business disruption.

3) *Goal of Release and Deployment:*

To deploy new releases into production, transition support to service operation, and enable its effective use in order to deliver maximum value to the clients.

4) *Objectives of Release and Deployment:*

a. To define and agree upon Release policies, and Release and Deployment plans with customers and stakeholders.
b. To ensure the integrity of constructed release packages and that they are recorded accurately in the Configuration Management System.
c. To ensure that all release packages can be tracked, installed, verified, uninstalled or backed out if necessary.
d. To ensure the required skills and knowledge is transferred to support team, customers, end users, suppliers and any other relevant stakeholders.
e. There is minimal unpredicted impact on the production services, customers and service operations.

5) *Scope of Release and Deployment:*

Release and Deployment works closely in conjunction with the other Release, Control and Validation (RCV) processes to enable the quality transition of services. The role played specifically by Release and Deployment is to build, package, validate and distribute authorized service changes to the target systems.

6) *Benefits of Release and Deployment:*

a. Delivering change, faster, at optimum cost and minimized risk
b. Assuring customers and users can use the new or changed service in a way that supports the business goals
c. Improving consistency in implementation approach across the business change, service teams, suppliers and customers
d. Contributing to meeting auditable requirements for traceability through Service Transition.

7) *Triggers and Interfaces*

The primary interfaces of release and deployment exist with Change Management and the surrounding Service Transition processes. Other inputs will also be provided from Service Strategy and Service Design to ensure that the requirements for value provision have been met.

The inputs to release and deployment include:

a. Authorized Request for Changes(RFCs)
b. Service Packages
c. Service Design Package
d. Service Acceptance Criteria
e. Service Management policies and standards
f. Build Models and plans
g. Exit and entry criteria for each stage of release and deployment

The outputs include:

a. Release and deployment plan
b. Updated RFCs for any required activities
c. Updated service catalogue reflecting any service changes
d. New or modified service
e. New or modified processes
f. Skilled and knowledgeable support staff
g. End users with capabilities to use the service
h. SLAs (Service Level Agreements), OLAs (Operational Level Agreements) , UCs(Underpinning Contract)
i. Deployment plans and packages
j. Service Transition Report

8) *Release Design Options*

When planning individual releases or defining the policies that should exist, consideration about the potential impact and need for resources will affect how releases will be deployed to the target locations. The common options for

deploying releases are described below:

a. Big Bang or Phased Approach
b. The Push or Pull Approach
c. Automated or Manual

Big Bang: Where the new or changed service is deployed to all user areas in one operation. This will often be used when introducing an application change and consistency of service across the organization is considered important.

Phased Approach: The service is deployed to a part of the user base initially, and then this operation is repeated for subsequent parts of the user base via a scheduled rollout plan. This will be the case in many scenarios such as in retail organizations for new services being introduced into the stores' environment in manageable phases.

The Push Approach: service component is deployed from the centre and pushed out to the target locations. In terms of service deployment, delivering updated service components to all users, either in big bang or phased forms is the use of a push approach, since the new or changed service is delivered into the users' environment at a time not of their choosing.

The Pull Approach: used for software releases. Software is made available in a central location but users are free to pull the software down to their own location at a time of their choosing or when a workstation restarts. The use of „Pull' updating a release over the internet has made this concept significantly more pervasive. A good example is virus signature updates, which are typically pulled down to update P.C's and servers when it best suits the customer; however at times of extreme virus risk this may be overridden by a release that is pushed to all know users.Pull approaches do not rest so heavily on accurate configuration data and they can trigger an update to user records. This may be through new users appearing and requesting downloads or expected users not doing so, triggering investigation into their continued existence.

Automation: Helps to ensure repeatability and consistency. The time required to provide a well-designed and efficient automated mechanism may not always be available or viable. Typical examples of activities that are capable of a degree of automation are:

a. Discovery tools aid release Planning Automate builds reduce time taken this in turn can resolve scheduling conflicts and delays
b. Automated configuration baselines procedures save time and reduce errors in identifying the status of CI's and releases during build, test and deployment etc

Manual: Important to monitor and measure the impact of many repeated manual activities as they are likely to be inefficient and error prone. This will ultimately slow down the release team and create resource and capacity issues that affect the agreed service levels

9) *Release Policy*

A Release Policy is the formal documentation of the overarching strategy for releases and was derived from the Service Design phase of the Service Lifecycle. It is the governing policy document for the process and must accommodate the majority of releases being implemented. Typical contents of a Release Policy include:

a. Level of infrastructure to be controlled by Releases
b. Preferred structure and schedules for Release Packages
c. Definition of major and minor releases, emergency fixes
d. Expected deliverables for each type of Release
e. Policy on the production and execution of back out plans
f. How and where Releases should be documented
g. Blackout windows for releases based on business or IT requirements
h. Roles and responsibilities defined for the Release and Deployment process, Supplier contacts and escalation points

10) *Release and Deployment Activities*

The Release Policy is the overarching strategy for Releases and was derived from the Service Design phase of the Service Lifecycle. The Release Plan is the operational implementation for each release. The Deployment Plan is the documented approach for distributing a single Release.

Steps for Release and Deployment Activities:

A. Release planning
B. Preparation for build, test and deployment
C. Build and test
D. Service test and pilots
E. Plan and prepare for deployment
F. Perform transfer. Deployment and retirement
G. Verify deployment
H. Early Life Support
I. Review and close the deployment
J. Review and close Service Transition

A. *Release Planning*

Any plans created for the release and deployment will need to be integrated with the overall Service transition plan, and conform to any policies that have been defined. For each release, plans should be authorized by Change Management and used to assist in the evaluation of the risk, impact and resource requirements for components of the change. Typically the release and deployment plans should document the:

a. Scope and content of the release
b. The risk assessment for the release
c. Affected stakeholders
d. Teams responsible for the release
e. Communication strategy to be used during the release and deployment process

Plans should take into account the acceptance criteria that exist for the release and when authorization points will

verify a pass or fail. The processes of Evaluation and Service Validation and Testing will be integrated here to assist in the determination whether to continue, pause or revert to previous stages of the release.

Build and test planning: The approach taken for building, testing and maintaining the controlled environments to production will need to be planned in order to enable optimum use of resources for the development of the release. The activities that occur here are:

a. Developing build plans based on the Service Design Package and defining any environment requirements.
b. Scheduling the resources and time required to setup the environments
c. Testing the build and compilation procedures
d. Scheduling the build and compilation activities
e. Assigning resources, roles and responsibilities for any key activities

Environments that may be utilized during this period include:

a. Build environments
b. Testing and integration environments
c. Deployment environments
d. Pilot environments

Utilizing Pilots: Pilots may be useful for testing the service with a group of participants that are representative of the broader end-user community. For this to be effective the scope needs to be carefully determined, as being either too large or too small will likely result in some negatively impact to the overall success and quality of the release and deployment process. Pilots should include mechanisms by which feedback can be gathered about various aspects of the release and associated processes. For complex releases or in large and diverse organizations it may be appropriate to use more than one pilot for address the different implementation and support issues that exist.

Deployment Planning: There are many factors that will be considered when choosing the most appropriate deployment strategy. Questions that must be answered include:

a. What needs to be deployed?
b. Who are the users?
c. Are there location dependences?
d. Where are the users?
e. Who else needs to be prepared well in advance?
f. When does the deployment need to be completed?
g. Why is the deployment happening?
h. What are the critical success factors and exit criteria?
i. What is the current capability of the service provider?

Financial/Commercial Planning: Where necessary, various financial and commercial factors will need to be assessed before the deployment activities, including:

a. Working capital
b. Contracts and licenses
c. Funding
d. Intellectual property requirements

B. *Preparation for build, test and deployment*

Before the actual building of the release occurs, the release design must be validated against the requirements defined for the new or changed service offering. This should be an independent evaluation that checks the release will deliver the predicted outcomes, and any issues documented in an interim evaluation report. Training of involved release and deployment staff. In many cases the introduction of a release may require additional training for the release, deployment, build and test teams. Such training could be related to the:

a. Service management processes to be used
b. Changes in security or health and safety procedures
c. Understanding of the Service Design documentation and plans
d. Technology being utilized for the release.

C. *Build and Test*

Wherever possible, repeatable practices and reusable components should be utilized to during the build and test of releases. This includes managing the:

1) Build, test and packaging environments
2) Compilation and packaging tools
3) Configuration of the releases themselves:
   a. Version control
   b. Documentation templates for testing and validation
   c. Access rights and security procedures

Release and build documentation: Documentation templates, procedures, knowledge bases and other guidance should be consistently available to support the release team in the activities performed. Typical documentation that will be used by the release teams include:

1) Contract agreements
2) Purchase requests
3) Health and Safety guidelines
4) Security policies
5) Licence agreements
6) Procedures for:
   a. Distributing software
   b. Delivering, moving and installing equipment
   c. Wiping sensitive data and media

d. Publishing, sharing and archiving knowledge, information and data.

Acquire and test required components: Release and Deployment should be interfaced with the organization's existing procurement processes to acquire any required components for the release. This will save time and effort in verifying assets, capturing and recording information, ensuring proof of licence and triggering updates to the Asset Management System. As part of the overall Service Validation and Testing, each of the individual components should be tested to verify that any quality criteria has been met, initiating action where quality criteria is not met.

Release Packaging: Build management procedures, tools and checklists should be utilized as part of the release packaging, to provide repeatable practices and expected outcomes. When a definitive package has been assembled, a baseline should be taken of the release package and the correct versioning and naming conventions applied.

Managing the build and test environments: The need for multiple environments in which to build and test will depend on the size, complexity, frequency and impact of the releases being managed. Test environments should be protected using a range of testing best practices, and appropriate access to these environments given based on the priorities defined. Automating the installation of systems and software reduces the workload of people, but also requires testing of the scripts and mechanisms that will be used.

D. *Service testing and pilots*

As part of a coordinated effort with Service Validation and Testing, testing and validation must be performed at multiple levels. With particular focus on the release itself, service rehearsals may be used, which simulates as much of the service as possible in an extensive and widely involved practice session. This would normally occur after other pilots have run, and is designed to be the last measure to detect any potential issues that will arrive during or after the deployment to the live environment.

Pilots: Previous planning should have already identified what pilots will be used as part of the overall release and deployment. Key actions to take during pilots are:

a. Training of any people involved
b. Documentation of any required procedures
c. Continual communication and engagement with customers and users
d. Determine the levels of support required for the actual release
e. Discover and fix issues wherever possible before the final deployment
f. Document improvements where appropriate and incorporate them into future plans

E. *Plan and prepare for deployment*

At this stage the focus is to prepare the organization and people for organizational change and to refine and deployment plans that have been documented. These plans should include guidance regarding:

a. Risk mitigation plans
b. Disposal and retirement plans
c. The logistics for delivery
d. Knowledge transfer
e. Mobilizing users to be ready to use the service
f. Mobilizing the support staff for service readiness

F. *Perform transfer, deployment and retirement*

During the actual implementation itself, the activities performed can be grouped under the following tasks:

a. Transfer financial assets
b. Transfer changes required to business/organization
c. Deploy processes and materials
d. Deploy Service Management Capability
e. Transfer service
f. Deploy service
g. Decommissioning and service retirement
h. Remove redundant assets

These activities will need to be modified to accommodate any items specified in the deployment plan as part of the acceptance criteria for go live'.

G. *Verify deployment*

Once the activities for the deployment of releases are finished, verification should occur that users are capable of operating the service. Verification should ensure that:

a. The service/release components are in place by means of a configuration audit
b. Documentation has been updated accordingly
c. Roles and responsibilities have been correctly assigned
d. The measurement and reporting systems are established to measure performance of the service

Any noted deviations from plans or other items raised should be documented in order to improve future implementations and processes used.

H. *Early Life Support*

The quality of transition to Service Operation is a crucial element to the success of the overall service change that is being implemented. Rather than simply hand off support post-deployment, the release and deployment teams should assist in managing any calls, incidents and problems that are detected in the early of the new or modified service. This

enables more stability in this vulnerable period, increased customer and user satisfaction, enhanced learning and better momentum for continual improvement. The resource allocation from these teams will then be gradually reduced while Service Operation takes on more responsibility for support.

The example shown in the figures above demonstrate how the number of incidents for deployment A was significantly reduced through the use of Early Life Support, including the training of users and staff, and the transfer of knowledge to service desk staff.

a. The acceptance criteria for finalizing Early Life Support should be agreed early in the process, which might include such conditions as: Users can use the service effectively and efficiently for their business activities
b. Service delivery is managed and controlled across any service provider interfaces
c. Service levels and performance standards are being consistently achieved
d. Stakeholder groups verify the transfer of required knowledge
e. All deliverables required for the release have been signed off.

I. Review and close the deployment

A formal review of deployments should be performed for all releases of a determined level of size, impact, cost or risk to the organization. The review seeks to ensure that all requirements for the release have been met and to identify and potential improvements that can be made. Items that should be reviewed include:

a. Any quality criteria deviations that exist. Any open actions or necessary fixes that have been identified
b. Review open changes
c. Review performance targets and achievements
d. Experiences and feedback from customers, users and staff involved with the deployment. All problems and known errors are documented and accepted by the business and/or suppliers
e. Check that any redundant assets have been removed

J. Review and close Service Transition

The final step required in finalizing the completion of Service Transition is a formal review appropriate to the relative scale of the change. This will utilize the process of Evaluation and is driven by Change Management, which will verify successful completion and that the handover to Service Operation is complete. The “lessons learnt” should be documented to provide any improvement opportunities that can be made and developed by Continual Service Improvement for future transitions.

III. Conclusion

This paper helps to understand the basics of ITIL framework and release methodology. Good implementation of ITIL in the business has many benefits such as a higher efficiency, a better overview and control of the business’ needs and many more. The study gives the high level understanding of how to release a change into the IT environment using ITIL efficiently for better benefits for the organizations.

IV. References

1) Rob Addy, Effective IT Service Management, Springer, ISBN 978-3-540-73197-9, 2007
2) L.Bass, M.Klein, and F.Bachmann, Quality Attribute Design Primitives and the attribute Driven design method, Proceeding of Fourth on Product Family Engineering, Bilbao, Spain, October 4, 2001
3) Office of Government Commerce (UK)CCTA and OGC. Retrieved May 5, 2005
4) www.ogc.uk
5) Majid Iqbal and Michael Nieves (2007). ITIL Service Strategy. The Stationery Office. ISBN 9780113310456
6) Vernon Lloyd and Colin Rudd, ITIL Service Design, The Stationery Office, ISBN 9780113310470, 2007
7) Shirley Lacy and Ivor Macfarlane . ITIL Service Transition. The Stationery Office, ISBN 9780113310487, 2007
8) David Cannon and David Wheeldon. ITIL Service Operation. The Stationery Office, ISBN 9780113310463, 2007.
9) Larry Klosterboer, Implementing ITIL Configuration Management, IBM Press, 2007
10) Hoek, A.Van Der, Hall, R. S., Heimbigner D., Wolf, A. L. ,Software release management, Proceedings of the 6th European conference held jointly with the 5th ACM SIGSOFT international symposium on Foundations of software engineering, p.159-175, September 22-25, Zurich, Switzerland, 1997.
11) Erenkrantz, J. R.Release Management within Open Source Projects. In: proceedings of the 3rd open source software development workshop. Portland, oregon, USA, may 2003, pp 51–55.
12) Annie Ibrahim Rana, Muhammad Waseem Arfi, "SOFTWARE RELEASE METHODOLOGY:A CASE STUDY", Student Conference on Engineering Sciences and Technology, 2005.
13) Tim Malone, Ivanka Menken and Gerard Blokdijk, "Release, Control and Validation",The Art of Service Pvt. Ltd.,2009.