Change and Release Management

Change vs. Release management

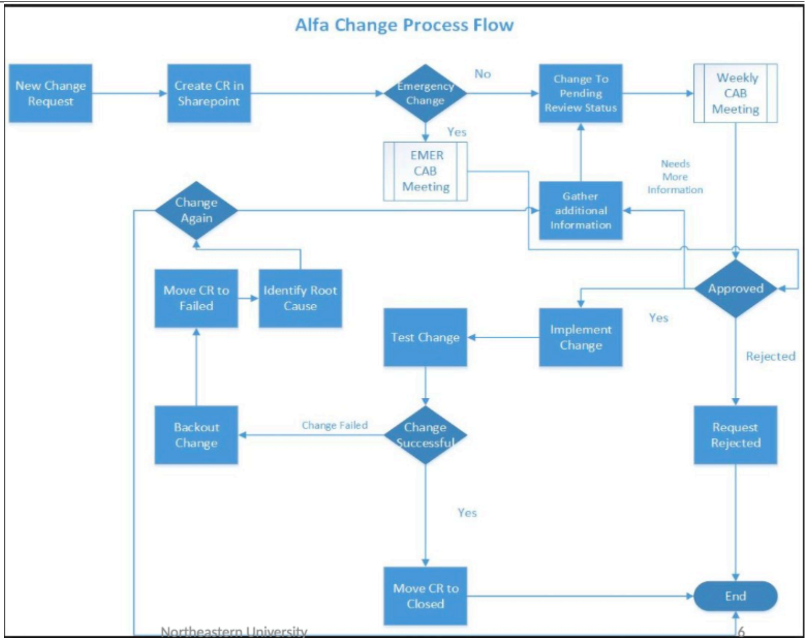
- What are Change & Release Management?
- What are their differences?
- What are their processes?
- ITIL Definitions and Services
- ISO 20000 Certification
- In production:
  - A defect is discovered
  - A new change is requested by the business

Change Management

- **ITIL Change Management:**
  - Definition: "It is the **quality control** process that sets the stage ready by assessing, planning and getting the right **approvals** for deployment of one or multiple changes to ensure **minimal disruption** to live environment".
  - Objective : The objective of **Change Management** is to ensure that **standardized** methods and procedures are used for efficient and prompt handling of all **changes** to control IT infrastructure, in order to **minimize** the number and impact of any related **incidents** upon service.
- ITIL stands for: Information Technology Information Library

Types of 'Changes'

1. Types of changes :
   1. Emergency Change/Urgent Change
   2. Standard Change
   3. Major Change
   4. Normal Change
2. Change Requests may include:
   1. Application Changes
   2. Hardware Changes
   3. Software Changes
   4. Network Changes
   5. Documentation Changes
   6. Environmental Changes
3. CAB: Change Advisory Board
4. CR: Change Request



The Definition of a "Release"

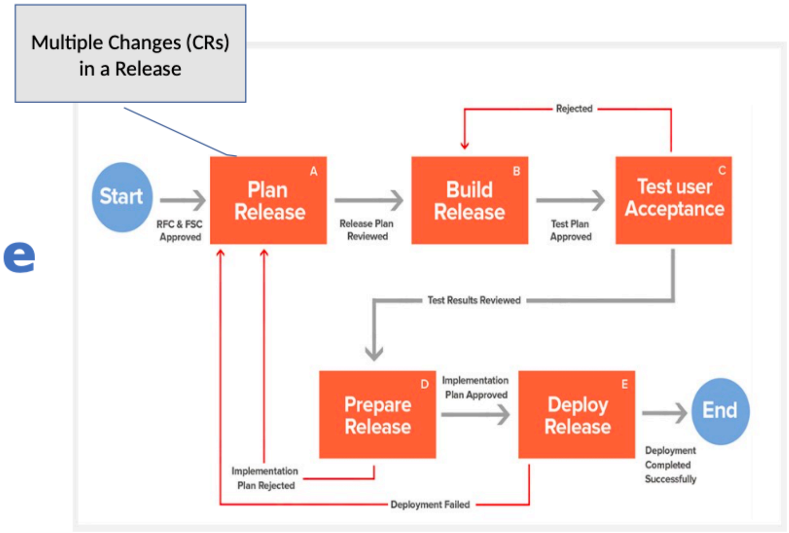
- A **release** (or a Software Release) is the distribution of the final version of an application.
- A software **release** may be either public or private and generally constitutes the initial generation of a **new** or **upgraded** application.
- Every **release** is comprised of a single **change** or multiple **changes.**

Release Management

1. **Definition: Release Management** is the process of **managing, planning, scheduling and controlling** a software build through different stages and environments; including **testing** and **deploying** software releases.
2. ITIL Release Management – "It takes care of the actual "**doing**" of deploying approved changes.

3. The **Release Manager** is responsible for implementing and **managing release** processes for code through **development, test, and production** environments.

Release Cycle



Change Management vs Release Management

Change Mangement	Release Managment
GateKeeper - protects the production environment while assessing the release plan	Doer - builds, tests and deploys changes as whole or in batch
Pre and post deployment activities	Deployment activities
Change schedule / forward schedule change .FSC	long-term release Windows
Not all changes result in a release	All release involve one or many changes
quality control point	Packing of approved changes
Authorization process	Implementation process
Strategic level	operation level
Post implementation review . PIR	version control

Release Manager Responsibilities

1. Drive **planning & delivery** of high-quality enterprise IT software releases.
2. Manage release schedules and milestones.
3. **Assemble and lead multiple cross-functional teams** to support IT application releases of varying size, complexity and duration.
4. **Collaborate** and manage **release processes** across all functions within IT and with business stakeholders.
5. **Manage change control** processes for the releases.
6. Implement **best practices** consistent with an agile development methodology.
   Proactively **identify and mitigate** risks and remove obstacles to release.
7. Guide the completion of root cause analysis (RCA) to ensure issues do not recur.
8. Perform **environment planning**, provisioning, and management required to support the releases.
9. Provide **oversight for deployment** of releases into production environments through ensuring release readiness and adequate deployment **runlist** handoff to Service Delivery.
10. **Communicate** plans and status, and escalate issues as required.

What is ITIL?

1. **ITIL stands for**: Information Technology Information Library
2. **Responsibility** :
    1. The *ITIL processes* within IT Service Management (ITSM) ensure that IT Services are provided in a focused, client-friendly and cost-optimized manner.
    2. With ITIL's help, IT Services are clearly **defined**, **success can be measured** with regards to the service provision, and targeted improvement measures can be introduced where necessary.
3. The **ITIL** processes are grouped into stages
    1. Service Strategy:
        1. **Process Objective:** To decide on a **strategy** to serve customers.
        2. **What to do:** Starting from an **assessment** of customer needs and the marketplace, the process determines which services the IT organization is to **offer** and what **capabilities** need to be **developed**.
        3. **Goal**: Its ultimate **goal** is to make the IT organization
        think and act in a **strategic** manner.
    2. Service Design
        1. **Process Objective:** To design new IT services.
        2. **Scope** : The scope of the process includes the **design** of new services, as well as **changes** and **improvements** to existing ones.
    3. Service Transition
        1. **Process Objective:** To build and deploy IT services.
        2. **Responsibility** : Service Transition also makes sure that changes to services and **Service Management** processes are carried out in a coordinated way.
    4. Service Operation
        1. **Process Objective:** To make sure that IT services are delivered **effectively** and **efficiently**.
        2. **Scope** :The Service Operation process includes **fulfilling** user requests, **resolving** service failures, fixing problems, as well as **carrying out** routine operational tasks.
    5. Continual Service Improvement (CSI)
        1. **Process Objective:** To use methods from quality management in order to learn from **past successes** and failures.
        2. **Goal**: The process aims to continually improve the **effectiveness** and **efficiency** of IT processes and services, in line with the concept of continual improvement adopted in ISO 20000.

- Each of the **five stages** is focused on a specific phase of a service's lifecycle.

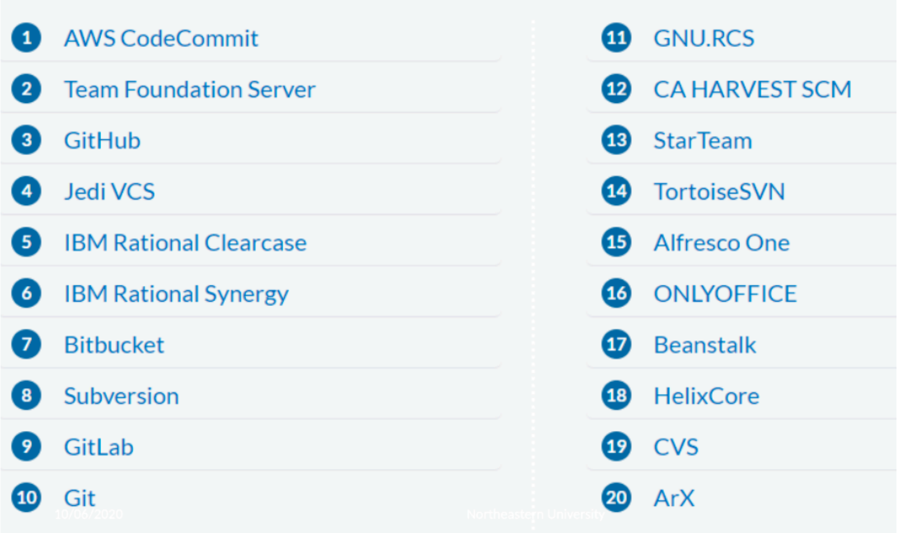
What is ISO?

1. **Stand for and definition**: The **International Organization for Standardization** is an international standard-setting body composed of representatives from various national standards organizations.
2. **Responsibility** : With a membership of 162 national standards bodies. Its members create international standards to make things work.
3. **What is ISO 20000?**
    1. ISO 20000 is the first ever international standard for IT service management.
    2. Its main focus is continual improvement of IT services.
    3. The standard works by setting **benchmarks** for companies based on evidence.
    4. **Achieving** these benchmarks proves that a company is capable of **consistent** excellence.
    5. The requirements include the **design, transition, delivery** and improvement of services.
    6. **Achievement** ISO 20000 certification shows an organizations **dedication** to an advanced IT Service Management approach.
4. **What is ISO9001?**
    1. **ISO 9001** is defined as the international standard that specifies requirements for a quality management system (QMS)
    2. The standard was most recently updated in 2015.
    3. Top Advantages:
        1. **Improves** the company credibility
        2. **Improves** customer satisfaction
        3. Create a culture of **continual improvement**
        4. **Engages** the company employees
5. **Who Can Use ISO Certification?**
    1. Organizations

        1. **Seeking services** from service providers and requiring assurance that their service requirements will be **fulfilled**.
        2. Require a consistent approach by all its service providers, including those in a supply chain.
    2. Service providers
        1. Intending to demonstrate their **capability** for the design, transition, delivery and improvement of services.
        2. Who monitor, measure and review its service management processes and services.
    3. Any IT service organization
        1. That wishes to demonstrate an advanced IT service management approach.

Version Control Systems

1. **Benefits**:
    1. The main advantages of using a **version control system:**
        1. **Streamlining** the development process,
        2. **Management** of code for multiple projects and **keeping a history** of all changes within a code.
    2. A version control software **saves** all the changes in a repository. Hence, if the developers make a mistake, **they can undo it**.
    3. At the same time, they can compare the new code with a previous version(s) to resolve their grievance.
    4. This can **reduce human errors** and **unintended consequences** to a great extent.
2. 20 best version control systems



1 AWS CodeCommit
2 Team Foundation Server
3 GitHub
4 Jedi VCS
5 IBM Rational Clearcase
6 IBM Rational Synergy
7 Bitbucket
8 Subversion
9 GitLab
10 Git
11 GNU.RCS
12 CA HARVEST SCM
13 StarTeam
14 TortoiseSVN
15 Alfresco One
16 ONLYOFFICE
17 Beanstalk
18 HelixCore
19 CVS
20 ArX

10/06/2020 Northeastern University

3. Example: CVS (concurrent version control system)
    1. CVS is one of the **oldest** version control system and is a well-known tool among both commercial and open source developers.
        1. It allows you to **check out** the code you are planning to work on, and **check-in** the changes.
        2. It has the capability to handle projects with multiple branches so that teams can **merge** their code changes and contribute unique features to the project.
        3. Since CVS is here for a long time now, it is the most mature version control software.