



University  
of Glasgow

## CSC2012 Professional Software Development 2 Lecture 01

# Project Management

Peter CY YAU

**WORLD  
CHANGING  
GLASGOW**

THE SUNDAY TIMES  
GOOD  
UNIVERSITY  
GUIDE  
2022

SCOTTISH  
UNIVERSITY  
OF THE YEAR



# Software Project Management

- Project management is needed as:
- Software development is subject to **budget** and **Schedule constraints**.
- Ensure delivering **high-quality** software.
- Software may be **delivered late**,
- **cost more** than originally estimated,
- or **fail to meet expectations** of customers.

# IEEE Software Life Cycle

- SQA – Software **quality assurance** • IEEE 730
- SCM – Software **configuration management** • IEEE 828
- STD – Software **test documentation** • IEEE 29119
- SRS – Software requirements specification • IEEE 29148
- V&V – Software **verification and validation** • IEEE 1012





## IEEE Software Life Cycle (Cont)

- SDD – Software design description • IEEE 1016
- SPM – Software project management • IEEE 16326
- SUD – Software user documentation • IEEE 24748
- SRA – Software reviews and audit • IEEE 1028



## Typical Pain Points

- Software product is **intangible**
- Cannot see **progress** by simply looking at **artefact** being constructed.
- Many software projects are “**one-off**” project
- Lessons learned from previous projects may not be **readily transferable** to new projects.
- Software processes are **variable & organization specific**
- Can't reliably **predict development problems**.



## Universal Management Activities

**Proposal writing** - Write a proposal to win a contract to carry out a work.

**Project planning** - Planning and scheduling project development.

**Risk management** - Assess risks, monitor these risks and take actions.

**People management** - Choose people and led to effective team performance.

**Reporting** - Report & communicate project progress, progress reviews.



## Triple Constraint

- The **project management triangle** (called also the **triple constraint**, **iron triangle** and **project triangle**) is a model of the constraints of project management.





## Triple Constraint Key Points

- The quality of work is constrained by the project's **budget (resources), deadlines (time) and scope (features)**.
- The project manager can **trade** between constraints.
- Changes in one constraint necessitate changes in others to **compensate** or **quality will suffer**.





# Common Software Management Activities

- Software project management comprises of a number of activities, which contains planning of project, deciding scope of software product, estimation of cost in various terms, scheduling of tasks and events, and resource management.
- Project management activities may include:
  - Project Planning
  - Scope Management
  - Project Estimation



## Pitfalls

- Planning **Assumptions**
- Make **realistic** rather than **optimistic** assumptions when defining a project plan.
- **Initial** assumptions and scheduling should take **unexpected problems into account**.
- Include in the plan, not seriously disrupted delivery schedule.

# Scope Management

- It defines the scope of project; this includes all the activities; process need to be done in order to make a deliverable software product.
- Scope management is essential because it creates **boundaries** of the project by **clearly defining what would be done** in the project and **what would not be done**.
- This makes project to contain **limited and quantifiable tasks**, which can easily be documented and in turn **avoids cost and time overrun**.



## Scope Management (Cont)

- During Project Scope management, it is necessary to
- Define the **scope**
- Decide its **verification and control**
- Divide the project into various **smaller parts** for ease of management.
- **Verify** the scope
- Control the scope by incorporating changes to the scope



# Project Estimation

- For an effective management accurate estimation of various measures is a must. With correct estimation managers can manage and control the project more efficiently and effectively.
- Project estimation may involve the following:
- Software size estimation
- Effort estimation
- Time estimation
- Cost estimation





## Software Size Estimation

- Software size may be estimated either in terms of **KLOC (Kilo Line of Code)** or by calculating number of function points in the software.
- **Lines of code** depend upon coding practices and Function points vary according to the user or software requirement.



## Effort Estimation

- The managers estimate efforts in terms of **personnel requirement and man-hour** required to produce the software.
- For effort estimation software size should be known. This can either be derived by managers' experience, organization's **historical data** or software size can be converted into efforts by using some **standard formulae**.

# Time Estimation

- Once size and efforts are estimated, the time required to produce the software can be estimated.
- Efforts required is **segregated into sub categories as per the requirement specifications and interdependency** of various components of software.
- Software tasks are divided into smaller tasks, activities or events by Work Breakthrough Structure (WBS). The tasks are scheduled on day-to-day basis or in calendar months.
- The sum of time required to complete all tasks in hours or days is the total time invested to complete the project.



## Cost Estimation

- This might be considered as the most difficult of all because it depends on more elements than any of the previous ones. For estimating project cost, it is required to consider
  - Size of software
  - **Software quality**
  - Hardware
  - Additional **software or tools, licenses, etc**
- Skilled personnel with task-specific skills
- Travel involved
- **Communication**
- Training and support



# Estimation Techniques

## Decomposition Technique

This technique assumes the software as a product of various compositions. There are two main models

Line of Code Estimation is done on behalf of number of line of codes in the software product.

Function Points Estimation is done on behalf of number of function points in the software product.



## Estimation Techniques (Cont)

### Empirical Estimation Technique

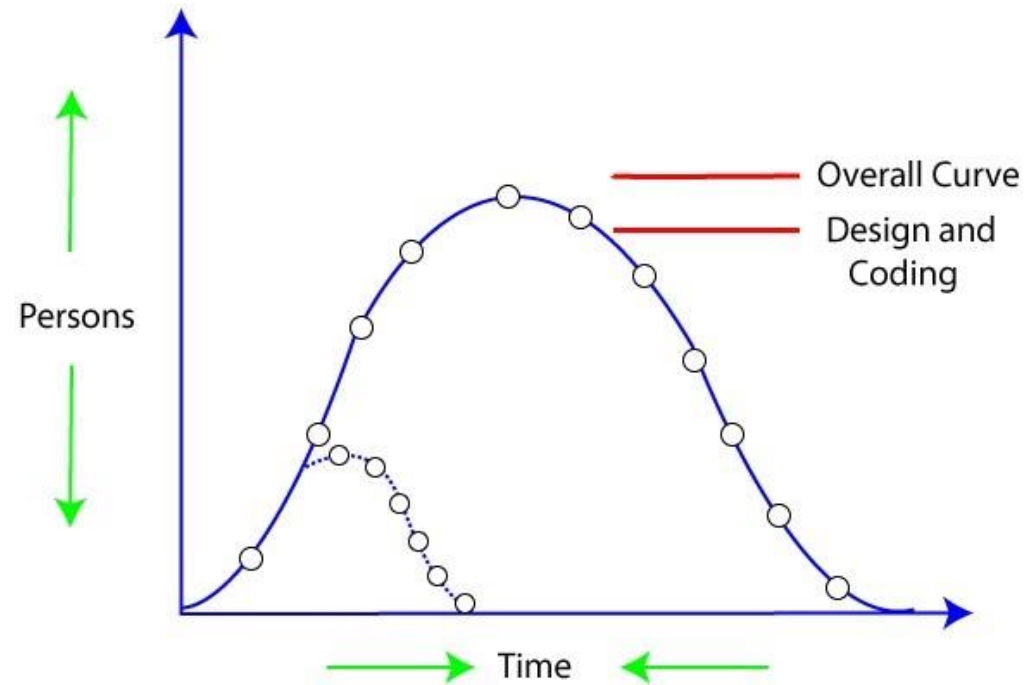
This technique uses empirically derived formulae to make estimation. These formulae are based on LOC or FPs.

**Putnam Model** - This model is made by Lawrence H. Putnam, which is based on Norden's frequency distribution (Rayleigh curve). Putnam model maps time and efforts required with software size.

**COCOMO** - COCOMO stands for **CO**nstructive **CO**st **MO**del, developed by Barry W. Boehm. It divides the software product into three categories of software: organic, semi-detached and embedded.



# Putnam Resource Allocation Model



The Rayleigh manpower loading Curve

[www.javatpoint.com/putnam-resource-allocation-model](http://www.javatpoint.com/putnam-resource-allocation-model)

## Putnam Resource Allocation Model (Cont)

- Putnam noticed that **software staffing profiles** followed the well known Rayleigh distribution. Putnam used his **observation about productivity levels** to derive the software equation:

$$L = C_k K^{1/3} t_d^{4/3}$$

- Putnam Resource Allocation Model proposed that optimal staff develop on a project should follow the Rayleigh curve.
- Only a **small number of engineers are required at the beginning of a plan** to carry out **planning and specification** tasks. As the project progresses and more detailed work are necessary, the number of engineers reaches a peak. After implementation and unit testing, the number of project staff falls.

[www.javatpoint.com/putnam-resource-allocation-model](http://www.javatpoint.com/putnam-resource-allocation-model)

# COCOMO Model

- Boehm proposed COCOMO (Constructive Cost Estimation Model) in 1981. COCOMO is one of the most generally used software estimation models in the world. COCOMO predicts the **efforts and schedule of a software product based on the size of the software**.

$$E = a * (KLOC)^b$$

1. Get an initial estimate of the development effort from evaluation of thousands of delivered **lines of source code (KDLOC)**.
2. Determine a set of **15** multiplying factors from various attributes of the project.
3. Calculate the effort estimate by multiplying the initial estimate with all the multiplying factors i.e., multiply the values in step 1 and step 2.



# Project Management Frameworks and Certificates

PMP: Project Management Practice

PRINCE2: PRojects IN Controlled Environments

ITIL: Information Technology Infrastructure Library



- PMP: Project Management Practice.
- Based on the principles set by the Project Management Institute (PMI).
- PMP is predominant in the US and is valued worldwide.
- PMP focuses more on the skills and knowledge required for the effective management of projects.
- Who is this for? Good for you if you are looking towards honing your skills, rather than learning about the framework of managing projects.

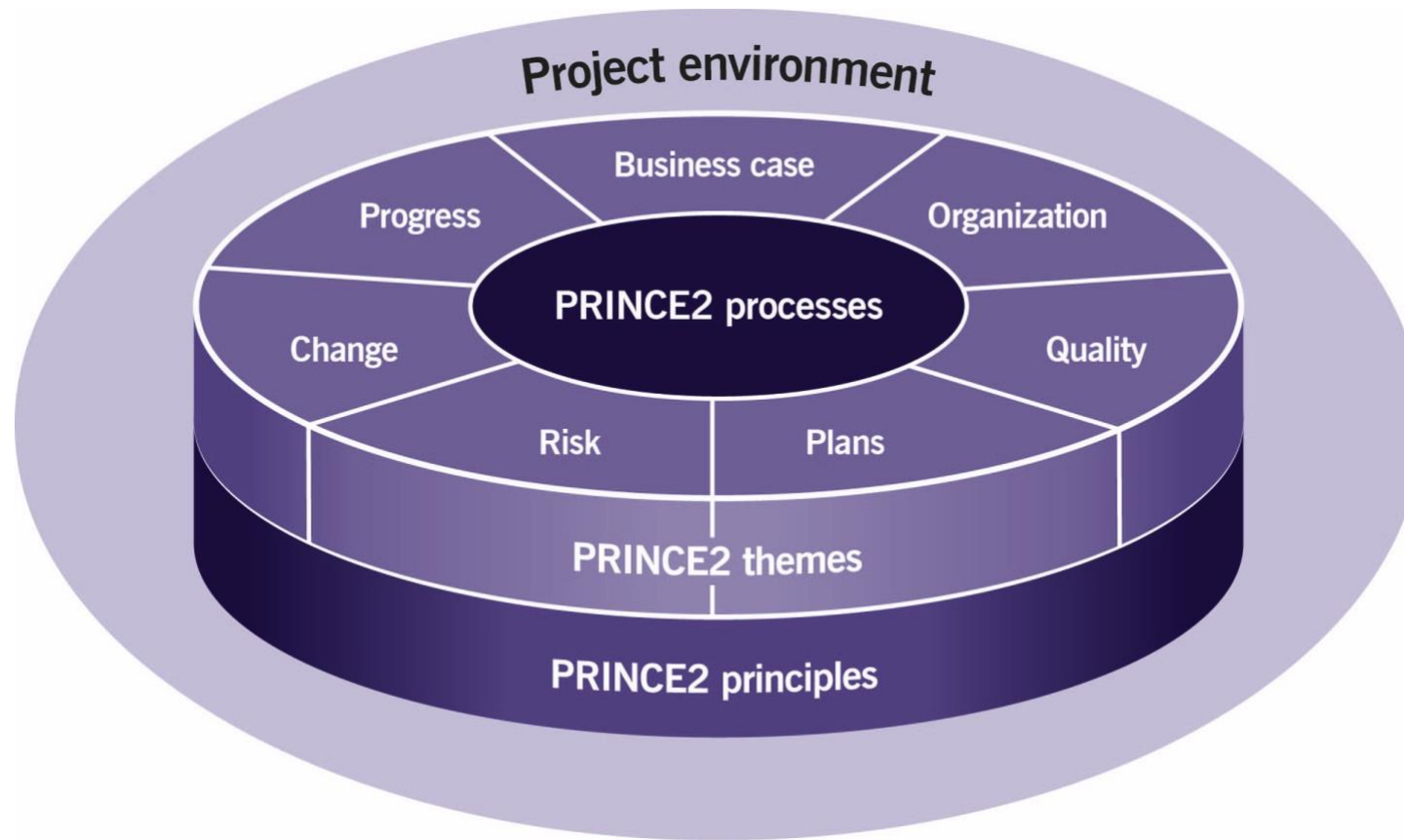
# PRINCE2

- PRINCE2: Abbreviated for PROjects IN Controlled Environments
- A **process-based methodology** that provides step-by-step and detailed procedures for delivering a successful project.
- It offers **clear instructions** and **steps and templates** for the users to follow so that the success of the project becomes inevitable.
- This certification is recognized and valued all over the world for its imperative design of managing projects and improving upon their **performance metrics**.
- Having quite a presence in the **European, Australian, and other Northern American** markets.

Source: <https://www.greycampus.com/blog/project-management/itil-prince2-or-pmp-which-is-right-for-you>

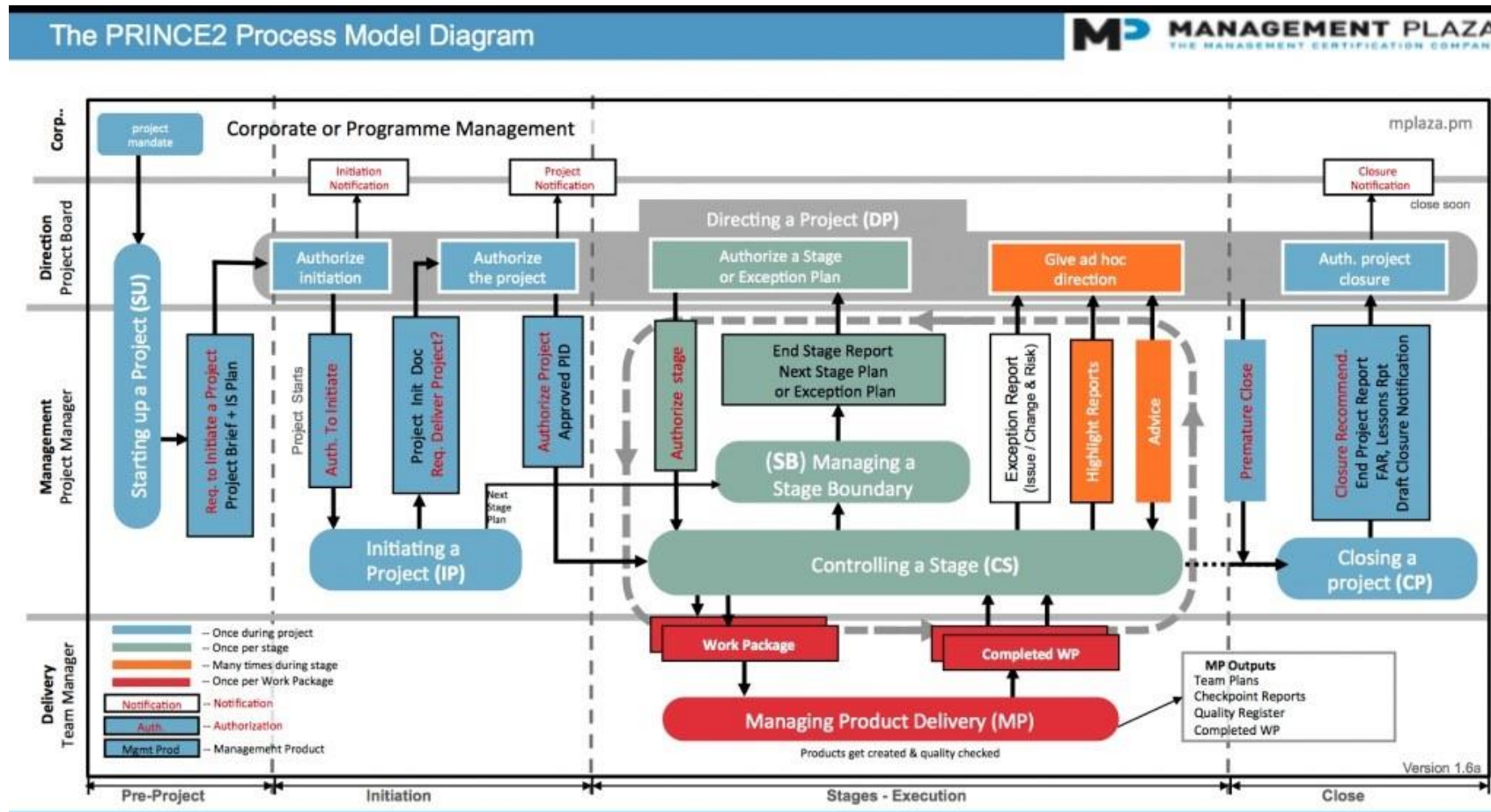


# PRINCE2 (Process) Visualization



[www.omnicom.digital/en/areas-of-interest/project-management-prince2-2/](http://www.omnicom.digital/en/areas-of-interest/project-management-prince2-2/)

# PRINCE2 Process Model Diagram



Source: <https://prince2.wiki/extras/prince2-process-model/>



## PRINCE2 Example

Tolerance type	Maintained in the project level	Example
Scope	Project plan	The printer should print the documents only in black/white.
Timescale	Project plan	The project/stage/team plan cannot last longer than 3 months.
Risk	Risk management approach	Printer might not work if it is exposed to water.
Quality	Project product description	Printer should not suffer mechanical failure for at least 10,000 pages.
Benefits	Business case	Printer should be set up by customer 10% faster than the ones from the competitors, and it must be 10% cheaper.
Cost	Project plan	The cost of the project should not exceed \$100,000.





## PRINCE2 Seven Principles (1,2)

- Continued Business Justification: The **business case** is the most important document and is **updated at every stage** of the project to ensure that the project is still **viable**. Early termination can occur if this ceases to be the case.
- Learn From Experience: Each project **maintains a lessons log** and projects should continually refer to their own and to previous and concurrent projects' lesson logs to avoid reinventing wheels. Unless lessons provoke change, they are only lessons identified (not learned).



## PRINCE2 Seven Principles (3,4)

**Defined Roles and Responsibilities:** Roles are separated from individuals, who may take on multiple roles or share a role. Roles in PRINCE2 are structured in four levels (corporate or programme management, project board, project manager level and team level). Project Management Team contains the last three, where all primary stakeholders (business, user, supplier) need to be presented.

**Manage by Stages:** The project is planned and controlled on a stage by stage basis. Moving between stages includes updating the business case, risks, overall plan, and detailed next-stage plan in the light of new evidence.

## PRINCE2 Seven Principles (5,6)

**Manage by Exception:** A PRINCE2 project has defined tolerances (6 aspects above) for each project objective, to establish limits of delegated authority. If a management level forecasts that these tolerances are exceeded (e.g. **time of a management stage will be longer than the estimated time in the current management stage**). It is escalated to the next management level for a decision how to proceed.

**Focus on Products:** A PRINCE2 project focuses on the definition and delivery of the products, in particular their quality requirements.



## PRINCE2 Seven Principles (7,8)

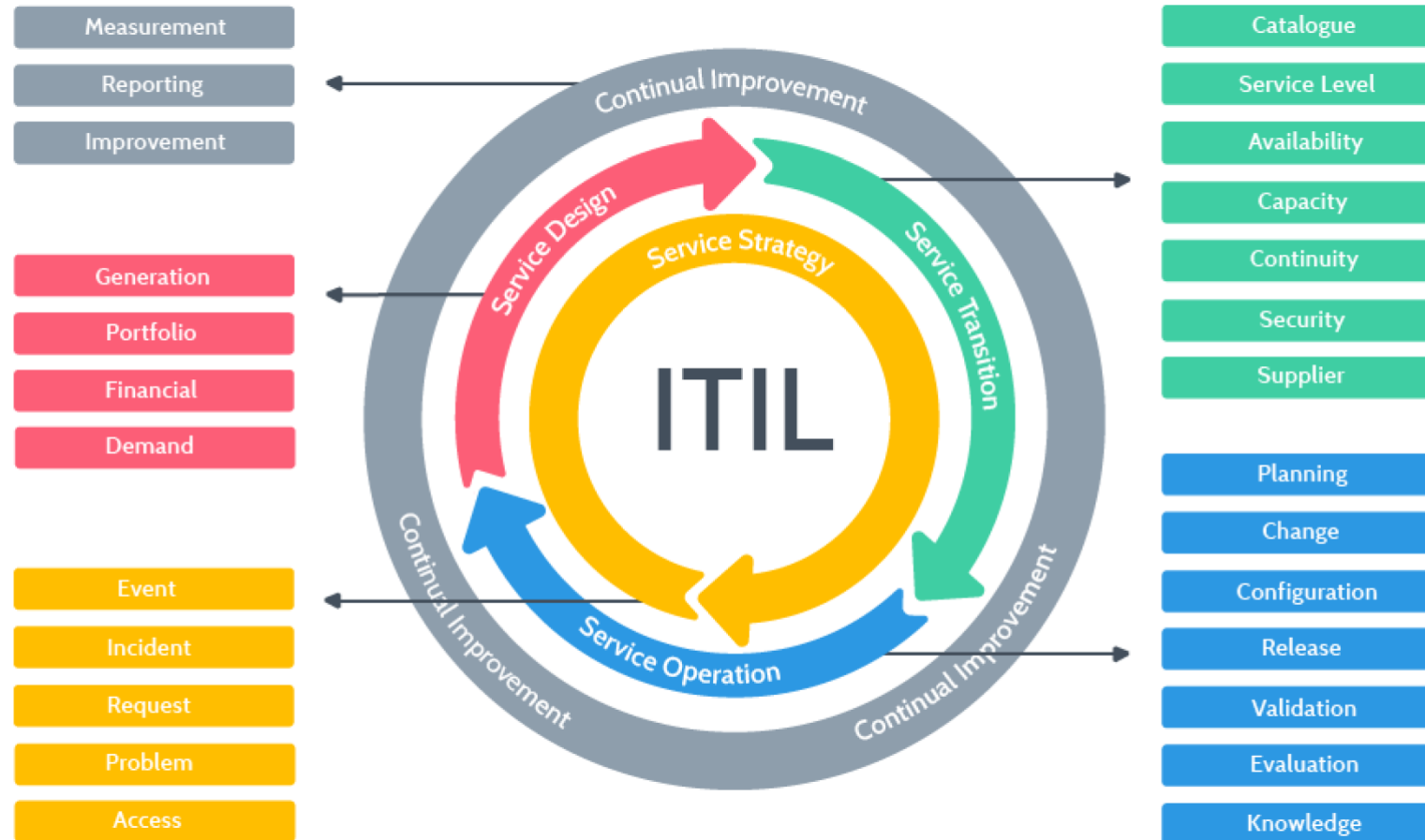
- **Tailor** to Suit Project Environment: PRINCE2 is tailored to suit the project environment, size, complexity, importance, time capability and risk. **Tailoring** is the first activity in the process Initiating A Project and reviewed for each stage.

- ITIL: stands for Information Technology Infrastructure Library
- Most recognized **framework for IT service management**.
- It is a process-based framework and enables an organization to improve and manage **IT services within the organizational system**.
- Who is this for? Globally accepted and helps professional in identifying various promising opportunities within the realm of **IT services**.
- What levels does it offer? Foundation, Intermediate, Managing Across the Lifecycle (MALC), Expert, and Master.

[www.greycampus.com/blog/project-management/itil-prince2-or-pmp-which-is-right-for-you](http://www.greycampus.com/blog/project-management/itil-prince2-or-pmp-which-is-right-for-you)



# ITIL Frameworks



[www.process.st/itil-processes/](http://www.process.st/itil-processes/)

## Acknowledgements

Teaching materials in this slide is derived, referenced and/or extracted from the following sources. We try to make the acknowledgements and references as accurate as possible. Thank you very much.

- Internet Resources, such as Wikipedia, Javapoint, Tutorialspoint, etc.
- IEEE Library, IEEE Xplore
- COMPSCI4015 Professional Software Development (H), University of Glasgow (UoG), Dr. Tim Storer
- COMPSCI3005 - Software Engineering M3, UoG, Dr. Richard McCreadie
- Software Engineering (Publisher: Pearson), Ian Sommerville
- Engineering Software Products: An Introduction to Modern Software Engineering (Publisher: Pearson), Ian Sommerville.
- Contents, and comments from Dr Cao Qi