

# Software Project Management 4th Edition



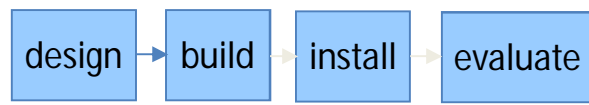
## Chapter 4

**Selection of an  
appropriate project  
approach**

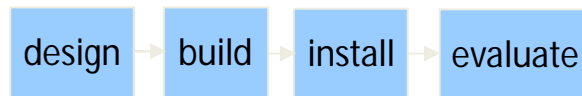
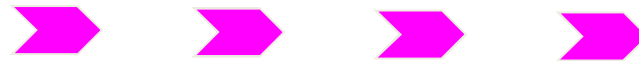
# Introduction

- Incremental delivery
- Benefits
- Agile Methods

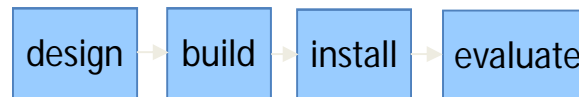
# Incremental delivery



first incremental delivery

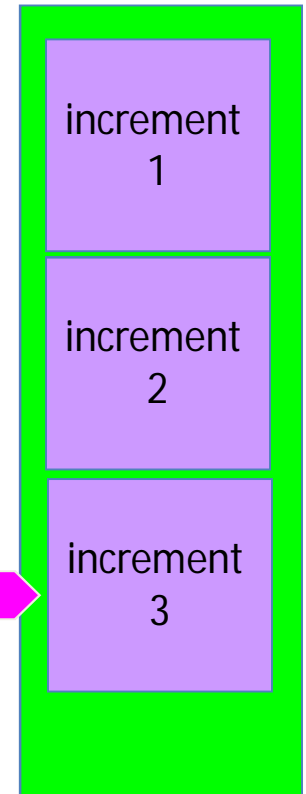


second incremental delivery



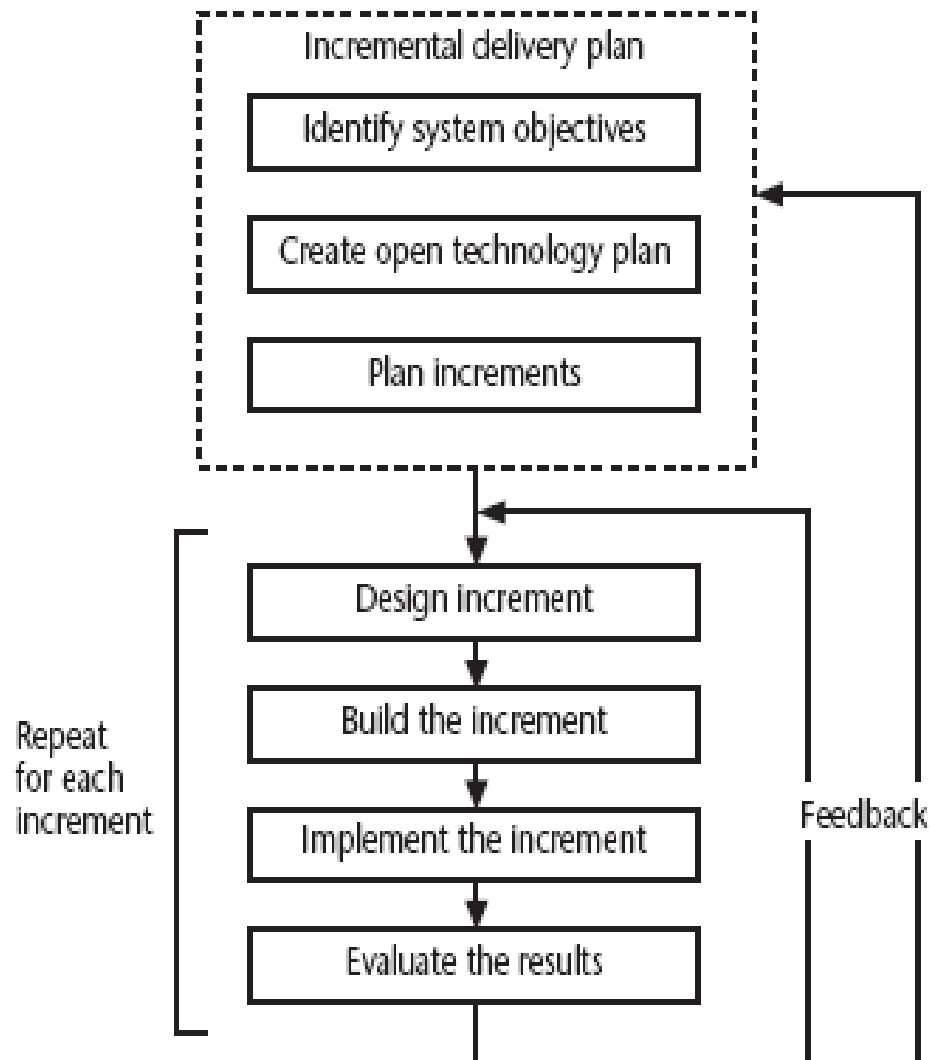
third incremental delivery

delivered system



# The incremental process

**Intentional  
incremental  
delivery**



# Incremental approach:benefits

- feedback from early stages used in developing latter stages
- shorter development thresholds
- user gets some benefits earlier
- project may be put aside temporarily
- reduces 'gold-plating':

BUT there are some possible disadvantages

- loss of economy of scale
- 'software breakage'

# The outline incremental plan

- steps ideally 1% to 5% of the total project
- non-computer steps should be included
- ideal if a step takes one month or less:
  - not more than three months
- each step should deliver some benefit to the user
- some steps will be physically dependent on others

# Which step first?

- some steps will be pre-requisite because of physical dependencies
- others may be in any order
- value to cost ratios may be used
  - $V/C$  where
  - $V$  is a score 1-10 representing value to customer
  - $C$  is a score 0-10 representing value to developers

# V/C ratios: an example

<b>step</b>	<b>value</b>	<b>cost</b>	<b>ratio</b>	
profit reports	9	1	9	2nd
online database	1	9	0.11	5th
ad hoc enquiry	5	5	1	4th
purchasing plans	9	4	2.25	3rd
profit- based pay for managers	9	0	inf	1st



# 'Agile' methods

structured development methods have some perceived advantages

- produce large amounts of documentation which can be largely unread
- documentation has to be kept up to date
- division into specialist groups and need to follow procedures stifles communication
- users can be excluded from decision process
- long lead times to deliver anything etc. etc

The answer? 'Agile' methods?

# Applications

- **Development Model**

Managing projects within the constraints of cost, schedule, functionality and quality is the biggest need of managing project successfully today.. [Website development solutions](#) have been developed during the past with important results published already some 15 years ago. Still, in practice not much has changed. The challenge is to find ways to catch the practical essence of solutions and ways to get the developers to use these solutions. Results: typically, projects can be done in 30% shorter time. While software development results were usually delivered late, the delays in other disciplines (like hardware and mechanical development) seemed to be non-existent.

# Research

- Incremental Delivery Through Continuous Design

In earlier Patterns in Practice columns, I've focused mainly on technical "patterns," but in this article I'll discuss the softer "practice" side of software design. The end goal of software projects is to deliver value to the customer, and my experience is that software design is a major factor in how successfully a team can deliver that value. Over design, under design, or just flat out wrong design impedes a project. Good design enables a team to be more successful in its efforts.

Reference Link : <http://msdn.microsoft.com/en-us/magazine/ee294453.aspx>