

Software Development Processes

13211239 YangBo*

ZJNU

Software Project Management

March 7, 2016

Abstract

The article is aim to show you about Software Development Processes(SDP) . the eassay display an overview of SDP and Step Wise project planning , besides ,there are some introductions to popular models of development approach so taht people can general understand how their work and the characters of the them .

keywords:project planning , software process models

1 Introduction

Software Development Processes(SDP) means the set of activities, methods, and practices that are used in the production and evolution of software . So in software engineering, we often use one specific embodiment of a software process architecture which is called software development methodology (also known as a system development methodology, software development life cycle, software development process, software process), is a splitting of software development work into distinct phases (or stages) containing activities with the intent of better planning and management. It is often considered a subset of the systems development life cycle. The methodology may include the pre-definition of specific deliverables and artifacts that are created and completed by a project team to develop or maintain an application.

2 Related Work

A variety of such frameworks have evolved over the years, each with its own recognized strengths and weak-

nesses. One software development methodology framework is not necessarily suitable for use by all projects. Each of the available methodology frameworks are best suited to specific kinds of projects, based on various technical, organizational, project and team considerations.

Software development organizations implement process methodologies to ease the process of development. Sometimes, contractors may require methodologies employed, an example is the U.S. defense industry, which requires a rating based on process models to obtain contracts. The international standard for describing the method of selecting, implementing and monitoring the life cycle for software is ISO/IEC 12207.(e.g., [1])

3 Overview

A process was defined as a collection of work activities, actions, and tasks that are performed when some work product is to be created. Each of these activities, actions, and tasks reside within a framework or model that defines their relationship with the process and with one another.Each software engineering action is defined by a task set that identifies the work tasks that are to be completed, the work products that will be produced, the quality assurance points that will be required, and the milestones that will be used to indicate progress.a generic process framework for software engineering defines five framework activities—communication, planning,modeling,construction,and deployment.In addition, a set of umbrella activities—project tracking and control, risk management, quality assurance, configuration management, technical reviews, and others—are

*e-mail:810847812@qq.com

The article is based on work to study Software Project Management.

Software Project Management (2016/17)

Author Name: 13211239 YangBo

University: ZJNU

Title: Software Development Processes

Supervisor: Dr. Kenwright

applied throughout the process.

4 Methods/Techniques

Common methodologies include waterfall, prototyping, iterative and incremental development, spiral development, rapid application development, extreme programming and various types of agile methodology.

- The Waterfall Model

The waterfall model, sometimes called the classic life cycle, suggests a systematic, sequential approach to software development that begins with customer specification of requirements and progresses through planning, modeling, construction, and deployment, culminating in ongoing support of the completed software. A variation in the representation of the waterfall model is called the V-model (Model 1).

- The V-model

Classic life cycle and the V-model. The V-model provides a way of visualizing how verification and validation actions applied to earlier engineering work. This model is the oldest paradigm for software engineering. However, over the past three decades, criticism of this process model has caused even ardent supporters to question its efficacy (Model 2).

- The incremental model: The incremental model combines elements of linear and parallel process flows. the incremental model applies linear in a staggered fashion as calendar time progresses. Each linear sequence produces deliverable “increments” of the software in a manner that is similar to the increments produced by an evolutionary process flow (Model 3)

- The Prototyping model

A customer defines a set of general objectives for software, but does not identify detailed requirements for functions and features. In other cases, the developer may be unsure of the efficiency of an algorithm, the adaptability of an operating system, or the form that human-machine interaction should take. In these, and many other situations, a prototyping paradigm may offer the best approach. Although prototyping can be used as a stand-alone process model, it is more commonly used as a technique that can be implemented within the context of any one (Model 4)

- A typical spiral model

Each of the framework activities represent one seg-

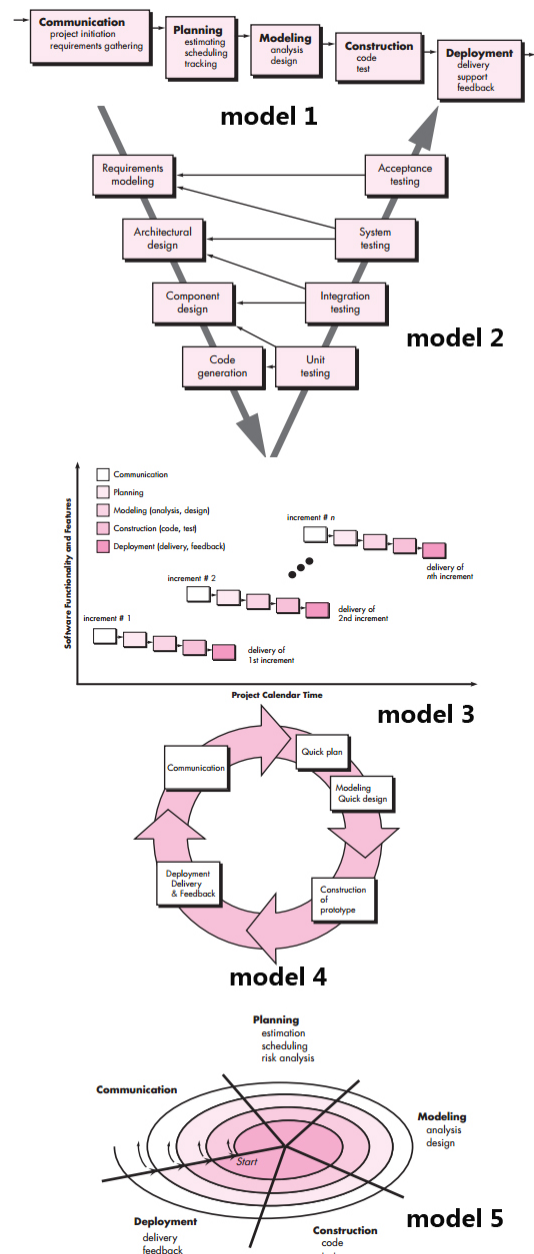


Figure 1: The Five Models

ment of the spiral path illustrated . As this evolutionary process begins, the software team performs activities that are implied by a circuit around the spiral in a clockwise direction, beginning at the center. Risk is considered as each revolution is made. Anchor point milestones—a combination of work products and conditions that are attained along the path of the spiral—are noted for each evolutionary pass. The first circuit around the spiral might result in the development of a product specification; subsequent passes around the spiral might be used to develop a prototype and then progressively more sophisticated versions of the software. (Model 5)

- The Concurrent Model

The concurrent development model, sometimes called concurrent engineering, allows a software team to represent iterative and concurrent elements of any of the process.

5 Conclusion

We can provide a common understanding and locate any inconsistencies, redundancies and omissions, besides, we also can reflect the development goals provide early evaluation, what's more, we will assist the development team to understand any special situation by using modelling. When you work to build a product or system, it's important to go through a series of predictable steps—a road map that helps you create a timely, high-quality result. The road map that you follow is called a “software process.”

References

- [1] From Wikipedia, the free encyclopedia
- [2] Software Project Management by Bob Hughes and Mike Cotterell