IT4460 - PHÂN TÍCH YÊU CẦU PHẦN MỀM

Năm học 2014-2015

Giảng viên: PGS. TS.Huỳnh Quyết Thắng
BM Công nghệ phần mềm
Viện CNTT-TT, ĐHBK HN
www.soict.hust.edu.vn/~thanghq

Chương 1. Tổng quan về YCPM và quy trình TKXDPM

I. Các khái niệm căn bản và định nghĩa

1. Các khái niệm và định nghĩa
2. Quá trình phát triển của phương pháp tiếp cận thiết kế và xây dựng phần mềm
3. Phân loại phần mềm theo ứng dụng
4. Quy trình phát triển phần mềm (SDLC)
5. Một số lưu ý liên quan đến YCPM

II. Phân loại các yêu cầu phần mềm

III. Các thuộc tính chất lượng của yêu cầu phần mềm

IV. Quy trình yêu cầu phần mềm: mô hình, tác nhân, hỗ trợ và quản lí, kết quả

V. Giới thiệu một số tài liệu liên quan đến nội dung chương

I. Các khái niệm căn bản và định nghĩa

1) Công nghệ phần mềm (Software Engineering)
2) Yêu cầu phần mềm (Software requirements)
3) Thiết kế phần mềm (Software design)
4) Xây dựng phần mềm (Software construction)
5) Kiểm thử phần mềm (Software testing)
6) Bảo trì phần mềm (Software maintenance)
7) Quản lý cấu hình phần mềm (Software configuration management)
8) Quản trị công nghệ phần mềm (Software engineering management)
9) Quá trình công nghệ phần mềm (Software engineering process)
10) Các công cụ và phương pháp trong CNPM (Software engineering tools and methods)
11) Chất lượng phần mềm (Software quality)

I. Các khái niệm căn bản và định nghĩa

1) Công nghệ phần mềm (Software Engineering)

Theo tổ chức IEEE trong "IEEE Standard Glossary of Software Engineering Terminology," IEEE, Piscataway, NJ std 610.12-1990, 1990.

Công nghệ phần mềm được định nghĩa (nguyên văn):

(1) The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software.

Từ khóa: hệ thống, khoa học, định lượng, phát triển, sử dụng và bảo trì phần mềm

(2) The study of approaches as in (1)

Từ khóa: nghiên cứu các hướng tiếp cận

I. Các khái niệm căn bản và định nghĩa

2) Yêu cầu phần mềm (Software requirements)

A requirement is defined as a property that must be exhibited in order to solve some real-world problem.

Từ khóa: property (thuộc tính/yêu cầu)

3) Thiết kế phần mềm (Software design) [IEEE610.12-90]:

Design is both "the process of defining the architecture, components, interfaces, and other characteristics of a system or component" and "the result of [that] process."

Từ khóa: process; kết quả của process; kiến trúc; thành phần; giao diện; các thuộc tính của hệ thống

4) Xây dựng phần mềm (Software construction)

Software construction refers to the detailed creation of working, meaningful software through a combination of coding, verification, unit testing, integration testing, and debugging.

I. Các khái niệm căn bản và định nghĩa

5) Kiểm thử phần mềm (Software testing)

Software Testing consists of the dynamic verification of the behavior of a program on a finite set of test cases, suitably selected from the usually infinite executions domain, against the expected behavior.

6) Bảo trì phần mềm (Software maintenance)

The maintenance phase of the lifecycle commences upon delivery but maintenance activities occur much earlier. Once in operation, anomalies are uncovered, operating environments change, and new user requirements surface.

I. Các khái niệm căn bản và định nghĩa

7) Quản lý cấu hình phần mềm (Software configuration management)

Software Configuration Management (SCM) is the discipline of identifying the configuration of software at distinct points in time for the purpose of systematically controlling changes to the configuration and of maintaining the integrity and traceability of the configuration throughout the system life cycle.

- *Management of the SCM process*. It covers the topics of the organizational context for SCM, constraints and guidance for SCM, planning for SCM, the SCM plan itself, and surveillance of SCM.

I. Các khái niệm căn bản và định nghĩa

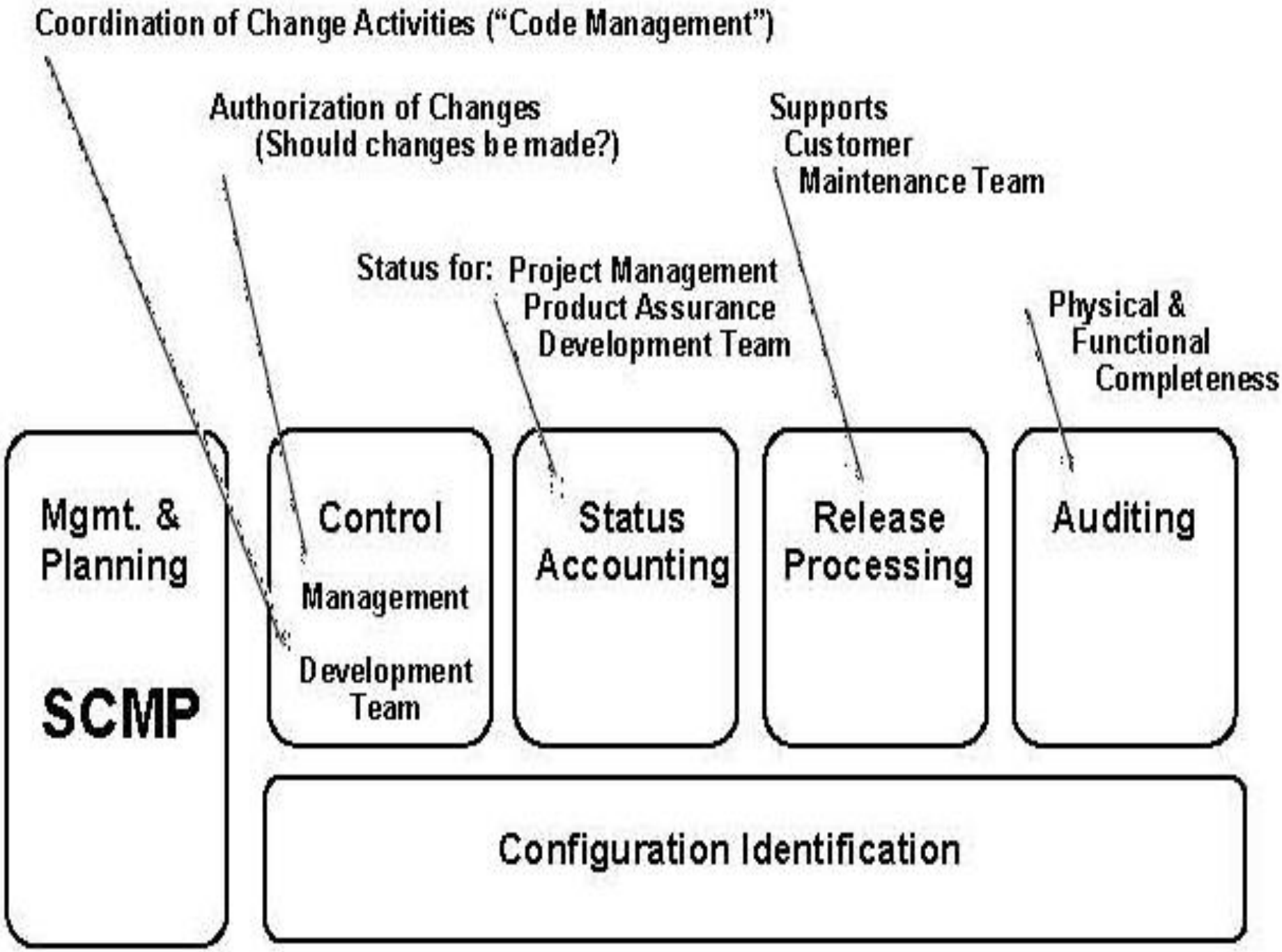
7) Quản lý cấu hình phần mềm (Software configuration management)

- *Software configuration identification*, which identifies items to be controlled, establishes identification schemes for the items and their versions, and establishes the tools and techniques to be used in acquiring and managing controlled items. The first topics in this sub-area are identification of the items to be controlled and the software library.
- *Software configuration control*, which is the management of changes during the software life cycle. The topics are: first, requesting, evaluating, and approving software changes; second, implementing software changes; and third, deviations, and waivers.

I. Các khái niệm căn bản và định nghĩa

7) Quản lý cấu hình phần mềm (Software configuration management)

- *software configuration status accounting*. Its topics are software configuration status information and software configuration status reporting.
- *software configuration auditing*. It consists of software functional configuration auditing, software physical configuration auditing, and in-process audits of a software baseline.
- *software release management and delivery*, covering software building and software release management.


Coordination of Change Activities ("Code Management")
Authorization of Changes
(Should changes be made?)
Supports
Customer
Maintenance Team
Status for: Project Management
Product Assurance
Development Team
Physical &
Functional
Completeness
Mgmt. &
Planning
SCMP
Control
Management
Development
Team
Status
Accounting
Release
Processing
Auditing
Configuration Identification

I. Các khái niệm căn bản và định nghĩa

8) Quản trị công nghệ phần mềm (Software engineering management)

- The Software Engineering Management addresses the management and measurement of software engineering. While measurement is an important aspect, it is here that the topic of measurement programs is presented. There are six sub-areas for software engineering management. The first five cover software project management and the sixth describes the software measurement programs.
- The first sub-area is *initiation and scope definition*, which comprises determination and negotiation of requirements, feasibility analysis, and process for the review and revision of requirements.

I. Các khái niệm căn bản và định nghĩa

8) Quản trị công nghệ phần mềm (Software engineering management)

- The second sub-area is *software project planning*, and includes process planning, determining deliverables, effort, schedule and cost estimation, resource allocation, risk management, quality management, and plan management.
- The third sub-area is *software project enactment*. The topics here are implementation of plans, supplier contract management, implementation of measurement process, monitor process, control process, and reporting.
- The fourth sub-area is *review and evaluation*, which includes the topics of determining satisfaction of requirements and reviewing and evaluating performance.

I. Các khái niệm căn bản và định nghĩa

8) Quản trị công nghệ phần mềm (Software engineering management)

- The fifth sub-area describes *closure*: determining closure and closure activities.
- Finally, the sixth sub-area describes *software engineering measurement*, more specifically, measurement programs.
- The topics of this sub-area are: establish and sustain measurement commitment, plan the measurement process, perform the measurement process, and evaluate measurement.

I. Các khái niệm căn bản và định nghĩa

9) Quá trình công nghệ phần mềm (Software engineering process)

The Software Engineering Process is concerned with the definition, implementation, assessment, measurement, management, change, and improvement of the software engineering process itself. It is divided into four subareas.

- The first sub-area presents *process implementation and change*. The topics here are process infrastructure, the software process management cycle, models for process implementation and change, and practical considerations.
- The second sub-area deals with *process definition*. It includes the topics of software life cycle models, software life-cycle processes, notations for process definitions, process adaptation, and automation

I. Các khái niệm căn bản và định nghĩa

9) Quá trình công nghệ phần mềm (Software engineering process)

- The third sub-area is *process assessment*. The topics here include process assessment models and process assessment methods.
- The fourth sub-area describes *process and product measurements*. The software engineering process covers general product measurement, as well as process measurement in general. The topics are process measurement, software product measurement, quality of measurement results, software information models, and process measurement techniques.

I. Các khái niệm căn bản và định nghĩa

10) Các công cụ và phương pháp trong CNPM Software engineering tools and methods

The Software Engineering Tools and Methods includes both software engineering tools and software engineering methods. The *software engineering tools* sub-area uses the same structure, with one topic for each of the other nine software engineering topics. An additional topic is provided: miscellaneous tools issues, such as tool integration techniques, which are potentially applicable to all classes of tools.

The *software engineering methods* sub-area is divided into four subsections: heuristic methods dealing with informal approaches, formal methods dealing with mathematically based approaches, and prototyping methods dealing with software development approachesbased on various forms of prototyping.

I. Các khái niệm căn bản và định nghĩa

11) Chất lượng phần mềm (Software quality)

The Software Quality deals with software quality considerations which transcend the software life cycle processes. Since software quality is a ubiquitous concern in software engineering, it is also considered in many of the other topics. The description of this topic covers three sub-areas.

- The first sub-area describes the *software quality fundamentals* such as software engineering culture and ethics, the value and costs of quality, models and quality characteristics, and quality improvement.

I. Các khái niệm căn bản và định nghĩa

11) Chất lượng phần mềm (Software quality)

- The second sub-area covers *software quality management processes*. The topics here are software quality assurance, verification and validation, and reviews and audits.
- The third and final sub-area describes *practical considerations* related to software quality. The topics are software quality requirements, defect characterization, software quality management techniques, and software quality measurement.

Chương 1. Tổng quan về YCPM và quy trình TKXDPM

I. Các khái niệm căn bản và định nghĩa

1. Các định nghĩa
2. Quá trình phát triển của phương pháp tiếp cận thiết kế và xây dựng phần mềm
3. Phân loại phần mềm theo ứng dụng
4. Quy trình phát triển phần mềm (SDLC)
5. Một số lưu ý liên quan đến YCPM

II. Phân loại các yêu cầu phần mềm

III. Các thuộc tính chất lượng của yêu cầu phần mềm

IV. Quy trình yêu cầu phần mềm: mô hình, tác nhân, hỗ trợ và quản lí, kết quả

V. Giới thiệu một số tài liệu liên quan đến nội dung chương

I. 2. Phương pháp tiếp cận PTTK phần mềm

- Hướng tiếp cận: Process-Oriented
  - Tập trung vào các giải thuật và thao tác xử lý dữ liệu
  - Quá trình phát triển phần mềm tập trung vào thể hiện các phương pháp xử lý dữ liệu
  - Cấu trúc dữ liệu thông thường không thể hiện rõ
  - Nhược điểm của hướng tiếp cận: Các tệp dữ liệu rất khó xây dựng để thoả mãn phần mềm

I. 2. Phương pháp tiếp cận PTTK phần mềm

- Hướng tiếp cận: Data-Oriented Approach
  - Mô tả tổ chức của dữ liệu, mô tả dữ liệu lấy ở đâu và sử dụng như thế nào
  - Mô hình dữ liệu được thành lập và được mô tả mối quan hệ giữa các dữ liệu tương ứng này và các quy định về mối quan hệ
  - Sử dụng các Business rules để chỉ ra phương pháp xử lý dữ liệu

I.2. Phương pháp tiếp cận PTTK phần mềm

- Hướng tiếp cận: Architecture-Oriented Approach
  - Lựa chọn kiến trúc và công nghệ phần mềm để thực hiện bài toán
  - Áp dụng các phương pháp Prototyping để nhanh chóng xây dựng được phần mềm
  - Sử dụng các Pattern kiến trúc mẫu để chỉ ra phương pháp xử lý dữ liệu

Chương 1. Tổng quan về YCPM và quy trình TKXDPM

I. Các khái niệm căn bản và định nghĩa

1. Các định nghĩa
2. Quá trình phát triển của phương pháp tiếp cận thiết kế và xây dựng phần mềm
3. Phân loại phần mềm theo ứng dụng
4. Quy trình phát triển phần mềm (SDLC)
5. Một số lưu ý liên quan đến YCPM

II. Phân loại các yêu cầu phần mềm

III. Các thuộc tính chất lượng của yêu cầu phần mềm

IV. Quy trình yêu cầu phần mềm: mô hình, tác nhân, hỗ trợ và quản lí, kết quả

V. Giới thiệu một số tài liệu liên quan đến nội dung chương

3. Phân loại phần mềm theo ứng dụng

- Phần mềm hệ thống (system)
- Phần mềm thời gian thực (real-time)
- Phần mềm quản lý (business)
- Phần mềm công nghệ và khoa học (engineering and scientific)
- Phần mềm nhúng (embedded)
- Phần mềm trí tuệ nhân tạo (AI)

Nguồn tài liệu: [1] Hoffer J. A. Modern System Analysis and Design. Third Edition. Addison-Wesley, 2004

Chương 1. Tổng quan về YCPM và quy trình TKXDPM

I. Các khái niệm căn bản và định nghĩa

1. Các định nghĩa
2. Quá trình phát triển của phương pháp tiếp cận thiết kế và xây dựng phần mềm
3. Phân loại phần mềm theo ứng dụng
4. Quy trình phát triển phần mềm (SDLC)
5. Một số lưu ý liên quan đến YCPM

II. Phân loại các yêu cầu phần mềm

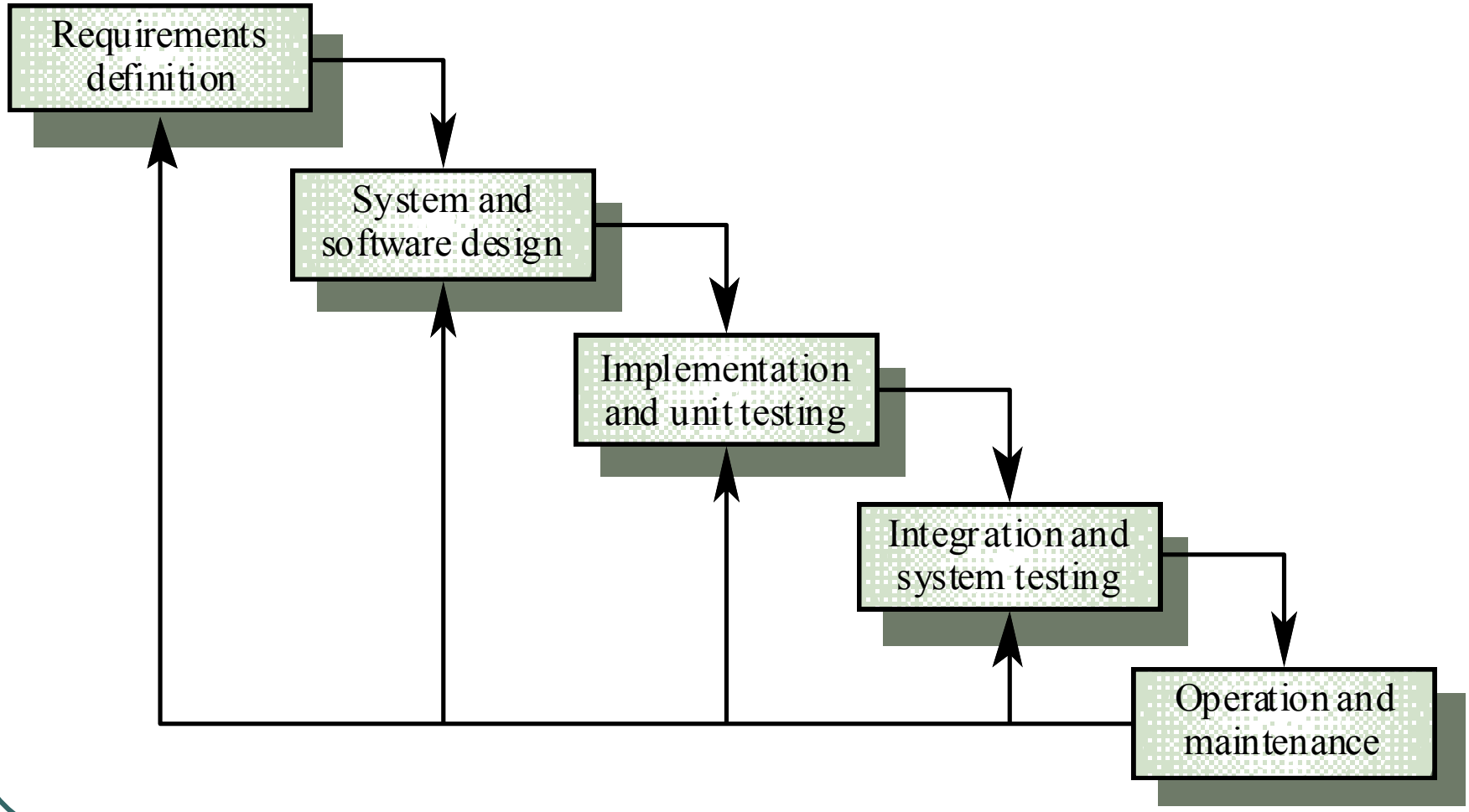
III. Các thuộc tính chất lượng của yêu cầu phần mềm

IV. Quy trình yêu cầu phần mềm: mô hình, tác nhân, hỗ trợ và quản lí, kết quả

V. Giới thiệu một số tài liệu liên quan đến nội dung chương

4. Quy trình phát triển phần mềm (SDLC)

- Mô hình thác nước (Waterfall model)



4. Quy trình phát triển phần mềm (SDLC)

- Mô hình thác nước

  (1) Phân tích yêu cầu phần mềm và đặc tả / Requirements analysis and definition

  (2) Thiết kế hệ thống và thiết kế phần mềm/System and software design

  (3) Thực thi viết mã nguồn và kiểm thứ khối/ Implementation and unit testing

  (4) Tích hợp và kiểm thử hệ thống/Integration and system testing

  (5) Sử dụng và bảo trì phần mềm / Operation and maintenance

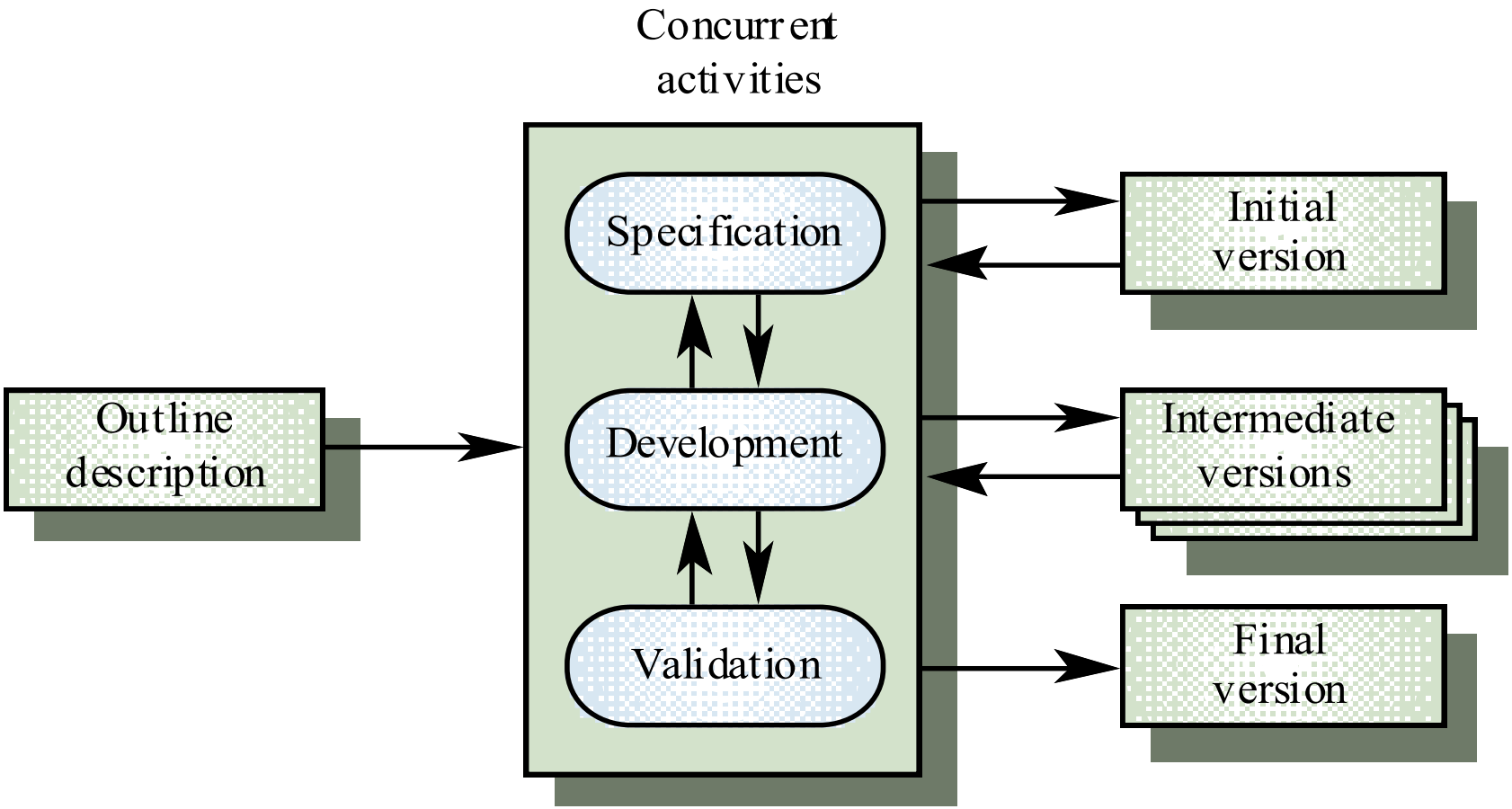
  Đặc điểm: Quay lại từ các giai đoạn sau trở về các giai đoạn trước để chỉnh sửa kết quả thực thi là không hiệu quả.

4. Quy trình phát triển phần mềm (SDLC)

- Các đặc điểm của mô hình thác nước
  - Inflexible partitioning of the project into distinct stages
  - This makes it difficult to respond to changing customer requirements
  - Therefore, this model is only appropriate when the requirements are well-understood

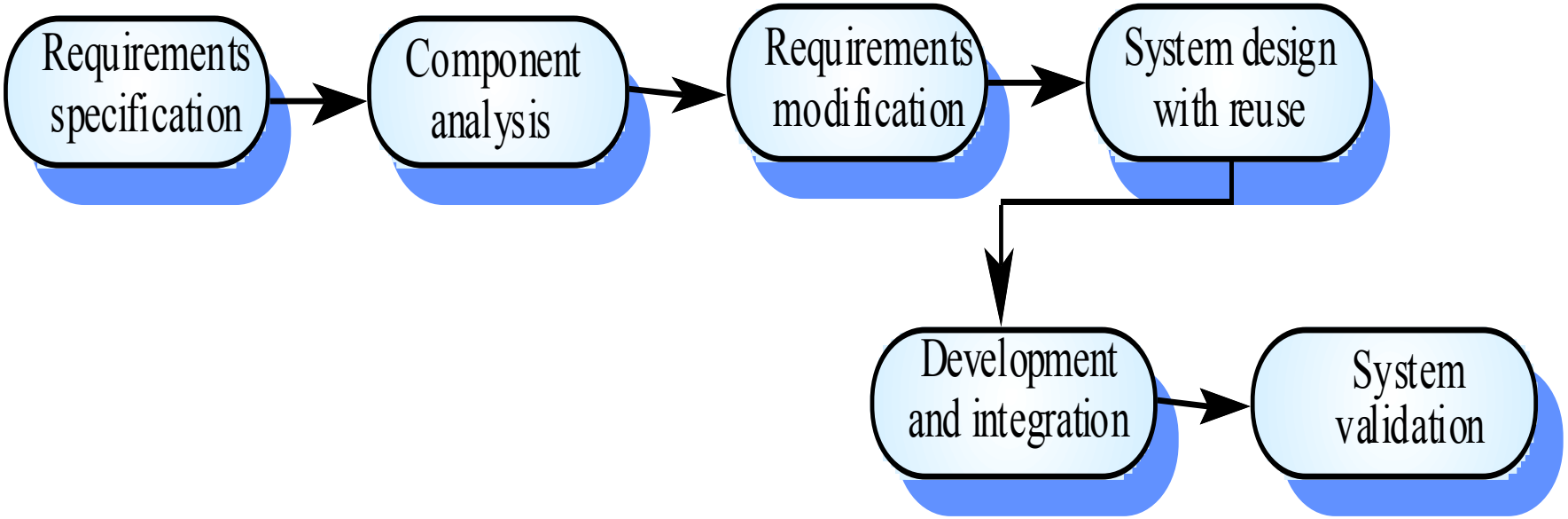
4. Quy trình phát triển phần mềm (SDLC)

Evolutionary development



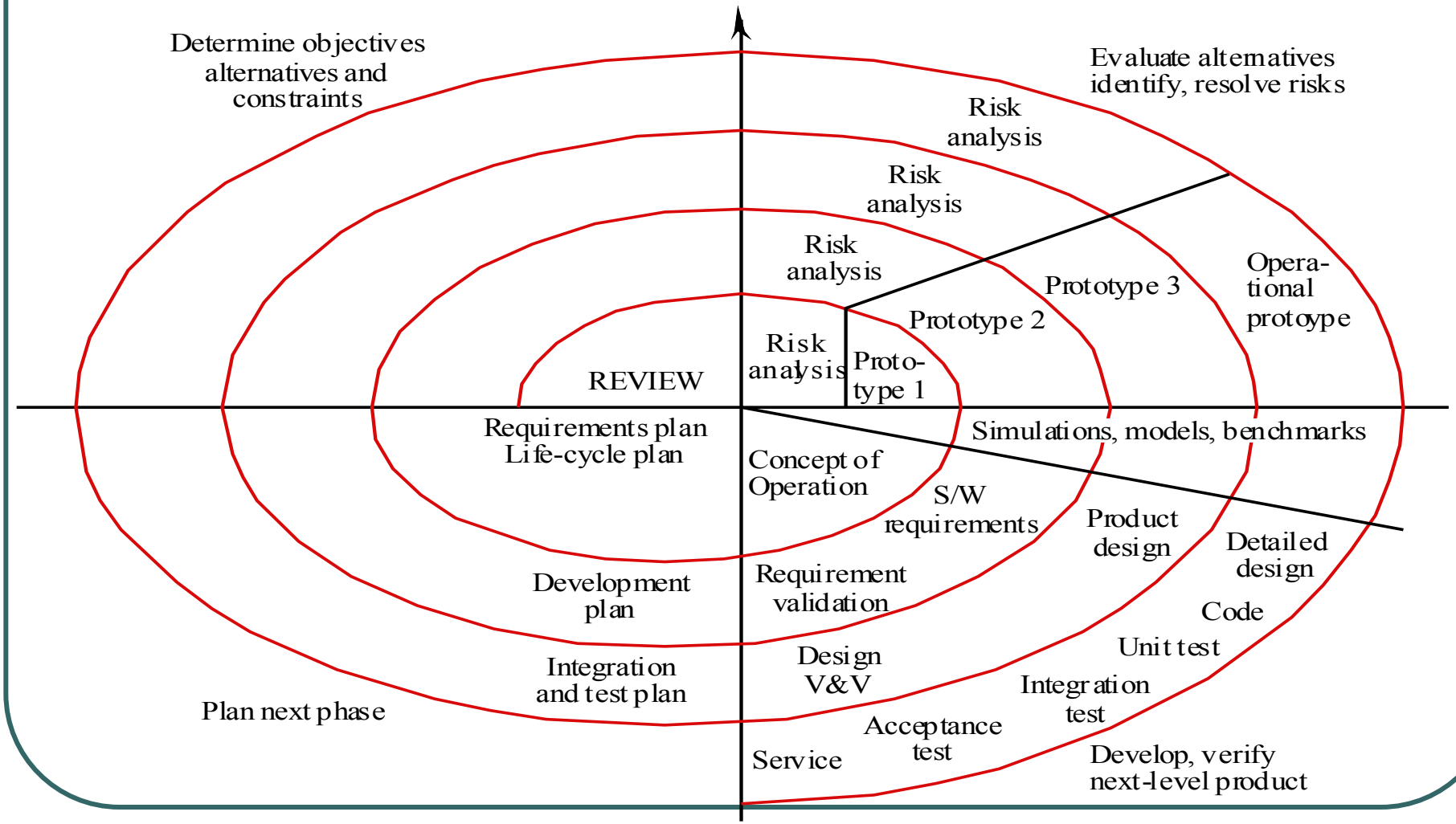
4. Quy trình phát triển phần mềm (SDLC)

Reuse-oriented development



4. Quy trình phát triển phần mềm (SDLC)

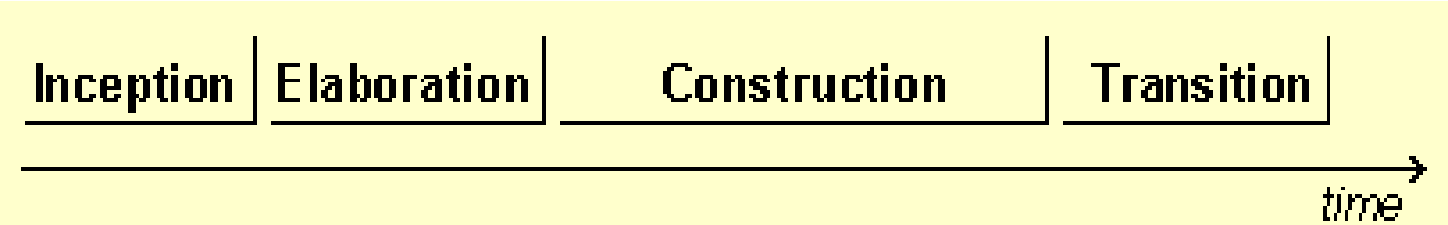
Spiral model of the software process



Iterative Model

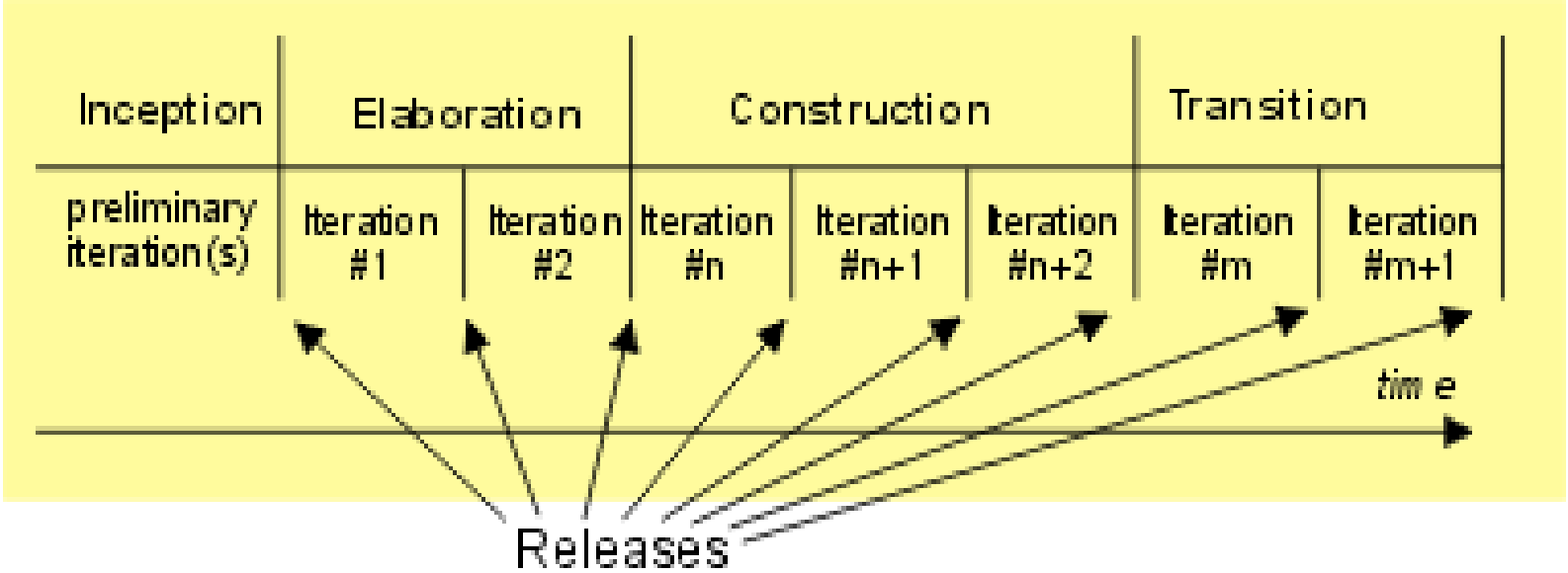
- According to Leffingwell
  - Four phases, Inception, Elaboration, Construction, Transition
  - Multiple iterations in each phase
  - Each iteration is a sequence of activities with a plan and evaluation criteria, resulting in an execution of some type
  - Each iteration builds on the functionality of the prior iteration

Iterative Lifecycle Phases



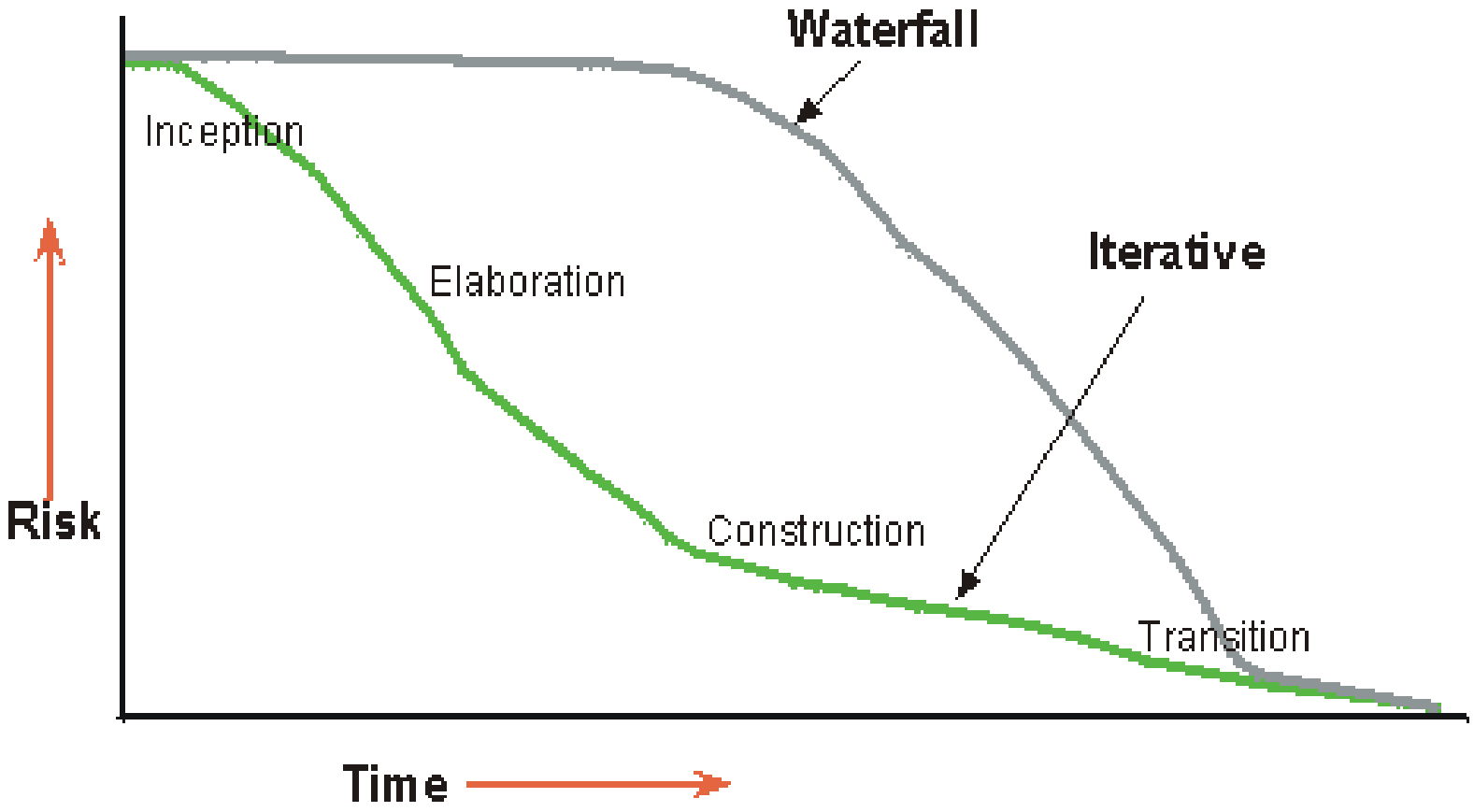
- Inception – Define the scope and business case of the project
- Elaboration – Refine the requirements, baseline architecture, build feasibility prototype
- Construction – Build the product
- Transition – Move the product into end-user community

Iterations and Phases

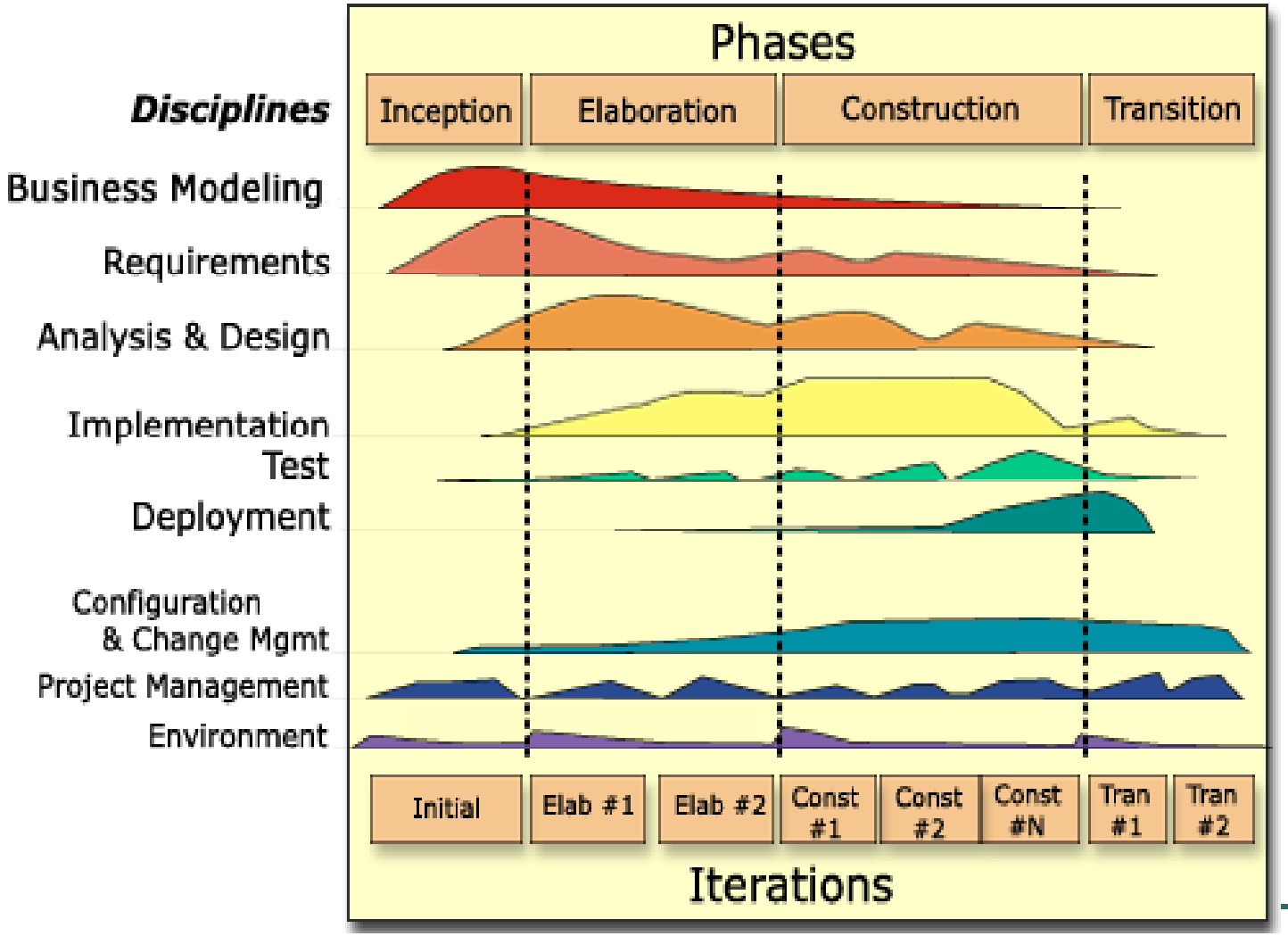


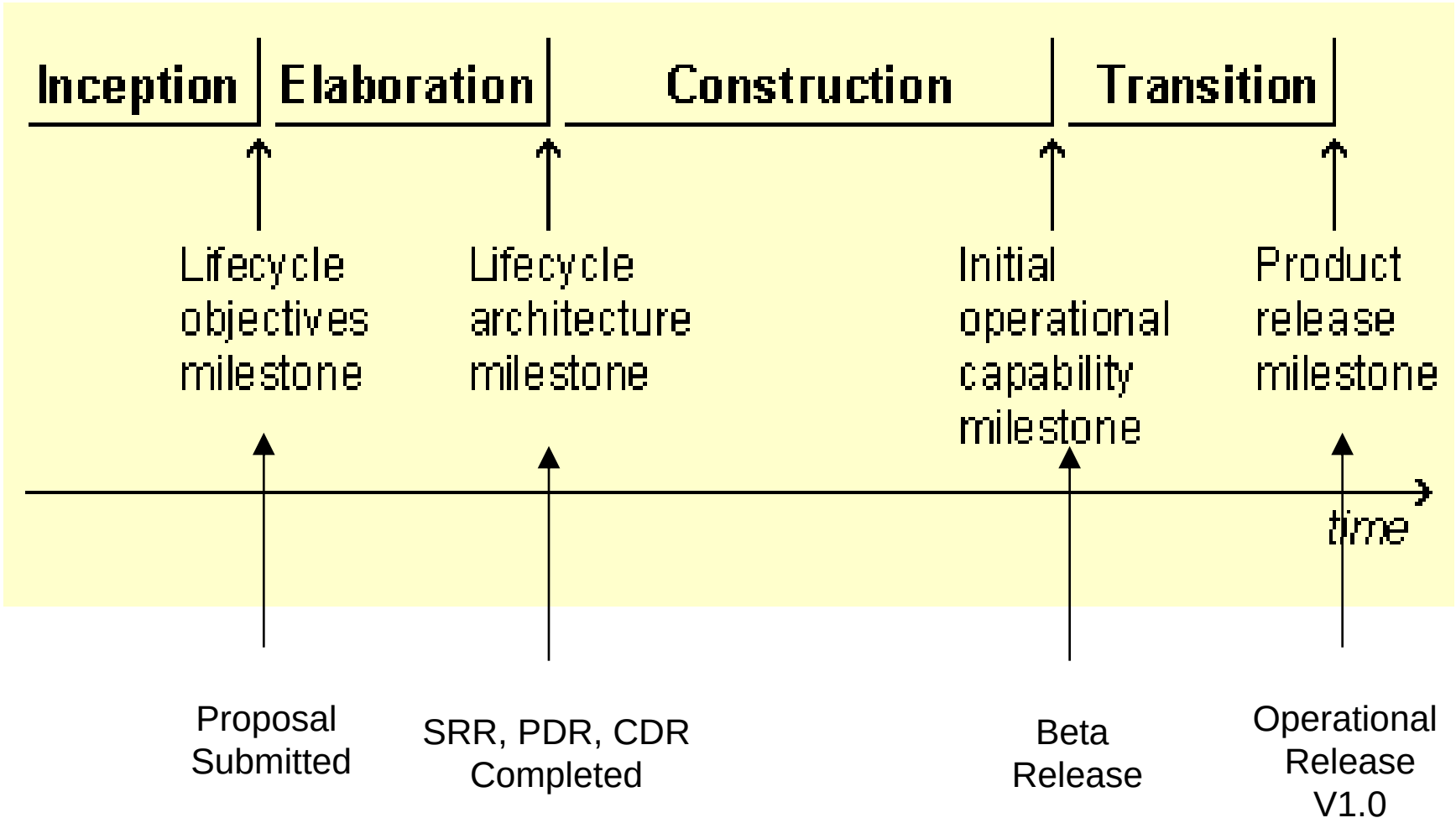
- An iteration is a distinct sequence of activities based on an established plan and evaluation criteria, resulting in an executable release (internal or external)

Risk Profiles



RUP's Iterative Approach





Mapping for a large development effort and the major milestones

Chương 1. Tổng quan về YCPM và quy trình TKXDPM

I. Các khái niệm căn bản và định nghĩa

1. Các định nghĩa
2. Quá trình phát triển của phương pháp tiếp cận thiết kế và xây dựng phần mềm
3. Phân loại phần mềm theo ứng dụng
4. Quy trình phát triển phần mềm (SDLC)
5. Một số lưu ý liên quan đến YCPM

II. Phân loại các yêu cầu phần mềm

III. Các thuộc tính chất lượng của yêu cầu phần mềm

IV. Quy trình yêu cầu phần mềm: mô hình, tác nhân, hỗ trợ và quản lí, kết quả

V.Câu hỏi và bài tập

VI. Giới thiệu một số tài liệu liên quan đến nội dung chương

VII. Nội dung đọc trước cho tuần sau

5. Một số nội dung cần lưu ý liên quan đến YCPM

- IEEE 1997 định nghĩa yêu cầu phần mềm:
  - (1) Điều kiện hay khả năng cần thiết để người sử dụng giải quyết được vấn đề hoặc mục tiêu hay đối tượng của họ
  - (2) Điều kiện hay khả năng được hiểu là đáp ứng của hệ thống hay thành phần hệ thống thoả mãn hợp đồng, tiêu chuẩn, đặc tả hoặc các văn bản bắt buộc khác.
  - (3) Văn bản thể hiện các điều kiện hoặc khả năng đã thể hiện ở (1) và (2).

5. Một số nội dung cần lưu ý liên quan đến YCPM

- Tính chất hai phía của Yêu cầu phần mềm và nhu cầu xác định điểm cân bằng
  - Trên thực tế đây một trong những vấn đề của công nghiệp phần mềm: thiếu sự định nghĩa chính xác về yêu cầu phần mềm.
  - Các yêu cầu xuất phát từ người sử dụng chương trình đối với người phát triển phần mềm thông thường quá là trừu tượng, ở mức cao.
  - Ngược lại những mô tả từ phía những người phát triển đối với người sử dụng lại ở mức quá thấp, quá chi tiết và người sử dụng lại không hiểu hết được.

5. Một số nội dung cần lưu ý liên quan đến YCPM

- Tính chất hai mặt của yêu cầu phần mềm: cách nhìn và suy nghĩ về phần mềm từ phía người sử dụng và cách nhìn và suy nghĩ về phần mềm từ phía người phát triển.
- Một trong những điều kiện quan trong nhất đã chỉ ra trong định nghĩa này là tính đặc tả của nó: phải được đúc kết lại bằng văn bản.
- Có sự tham gia và xác nhận của cả hai phía: người sử dụng va người phát triển
- Quá trình phát triển về cách hiểu của yêu cầu phần mềm sẽ cho thấy tính đầy đủ của định nghĩa thêo IEEE.

5. Một số nội dung cần lưu ý liên quan đến YCPM

- Các định nghĩa khác về yêu cầu phần mềm:
  - Alan Davis (1993): các nhu cầu của người sử dụng (các đặc tính, chức năng và đặc điểm) của hệ thống cần phải thể hiện từ các quan sát bên ngoài vào hê thống.
  - Jones (1994): Các yêu cầu và phát biểu của người sử dụng khởi sự quá trình phát triển phần mềm.
  - Sommervile và Sawyer (1997): Yêu cầu của người sử dụng là các đặc tả mô tả cần phải làm cái gì. Các đặc tả này giải thích các thuộc tính các đặc trưng của hệ thống và các hoạt động chức năng của hệ thống.

Chương 1. Tổng quan về YCPM và quy trình TKXDPM

I. Các khái niệm căn bản và định nghĩa

1. Các định nghĩa
2. Quá trình phát triển của phương pháp tiếp cận thiết kế và xây dựng phần mềm
3. Phân loại phần mềm theo ứng dụng
4. Quy trình phát triển phần mềm (SDLC)
5. Một số lưu ý liên quan đến YCPM

II. Phân loại các yêu cầu phần mềm

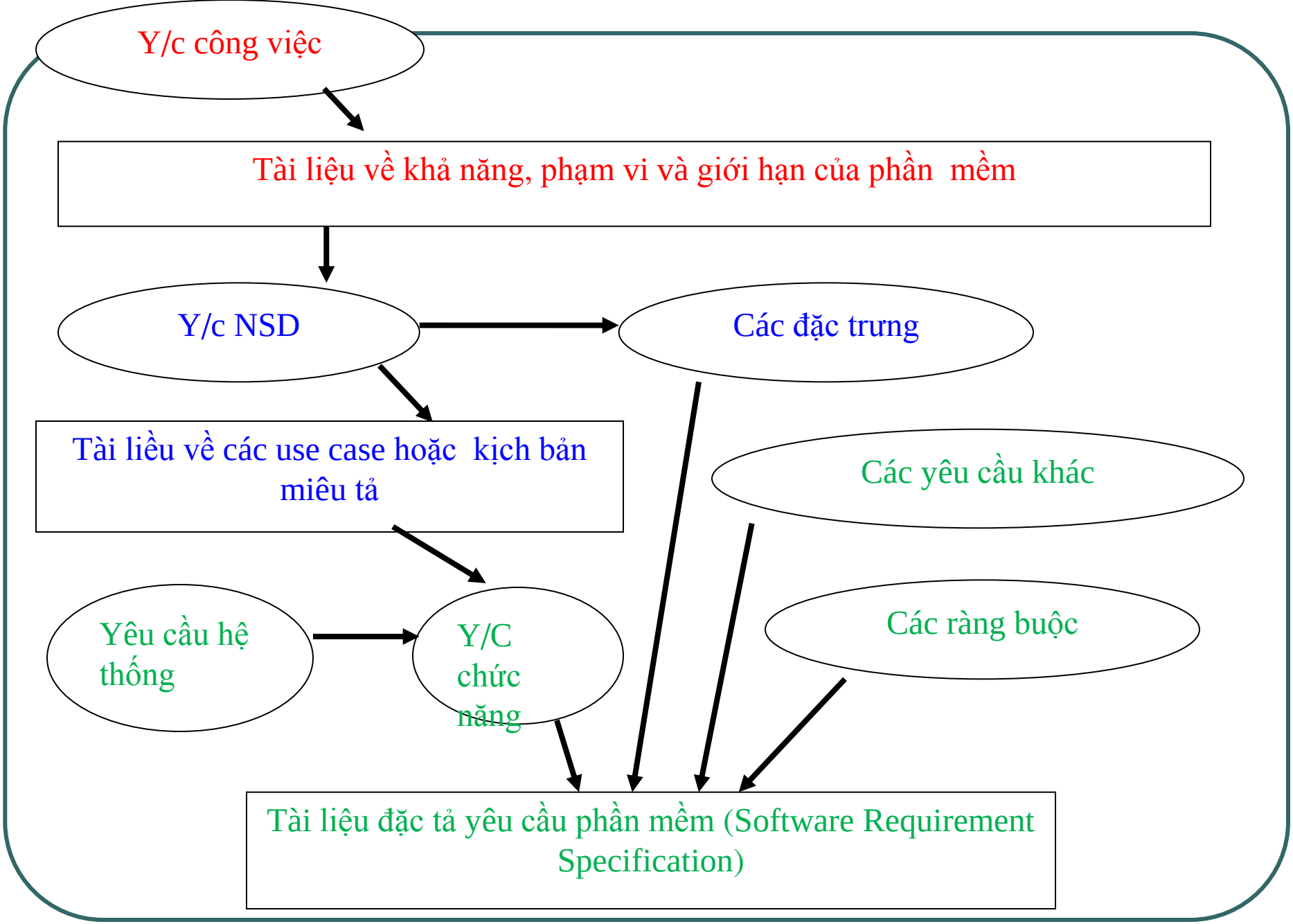
III. Các thuộc tính chất lượng của yêu cầu phần mềm

IV. Quy trình yêu cầu phần mềm: mô hình, tác nhân, hỗ trợ và quản lí, kết quả

V.Câu hỏi và bài tập

VI. Giới thiệu một số tài liệu liên quan đến nội dung chương

VII. Nội dung đọc trước cho tuần sau


Y/c công việc
Tài liệu về khả năng, phạm vi và giới hạn của phần mềm
Y/c NSD
Các đặc trưng
Tài liều về các use case hoặc kịch bản miêu tả
Các yêu cầu khác
Yêu cầu hệ thống
Y/C chức năng
Các ràng buộc
Tài liệu đặc tả yêu cầu phần mềm (Software Requirement Specification)

II. Phân loại yêu cầu phần mềm

- **Các mức độ (level) của yêu cầu**
  - Mức độ 1: *Yêu cầu công việc* (Business Requirement): thể hiện các mục tiêu yêu cầu ở mức cao của tổ chức hay khách hàng về khả năng, phạm vi ứng dụng và giới hạn của phần mềm
  - Mức độ 2: *Yêu cầu người sử dụng* (user requirement): thể hiện các nhiệm vụ cụ thể mà NSD cần phải hoàn thành, làm được với phần mềm. Thông thường được miêu tả bằng các trường hợp sử dụng (user case) hoặc các kịch bản miêu tả (scenario description).

II. Phân loại yêu cầu phần mềm

- **Các mức độ (level) của yêu cầu**
  - Mức độ 3: *Yêu cầu chức năng* (Functional Requirement): định nghĩa các chức năng phần mềm mà người phát triển cần phải xây dựng để có thể đáp ứng được tất cả các nhiệm vụ đã miêu tả trong yêu cầu người sử dụng ở mức trên.
  - Các yêu cầu chức năng được văn bản hóa bằng một tài liệu: Tài liệu đặc tả yêu cầu phần mềm (Software Requirement Specification)

II. Phân loại yêu cầu phần mềm

- **Các tài liệu trong yêu cầu phần mềm**
  - Mức độ 1: Tài liệu về khả năng, phạm vi và giới hạn của phần mềm
  - Mức độ 2: Tài liều về các use case hoặc kịch bản miêu tả
  - Mức độ 3: Tài liệu đặc tả yêu cầu phần mềm (Software Requirement Specification): tài liệu đặc tả mô tả đầy đủ các hoạt động, chức năng cần phải có của phần mềm. Đối với các hệ thống phần mềm lơn đây được coi là một thành phần của hệ thống sẽ chuyển giao cho khách hàng

II. Phân loại yêu cầu phần mềm

- **Các điểm lưu ý trong phân loại yêu cầu phần mềm**
  - Các yêu cầu khác: (nonfunctional requirement): thông thường chứa các chuẩn, các điều chỉnh, các điều kiện mà phần mềm cần phải có
  - Các ràng buộc: (constraint): các quy định chặt chẽ mà các phân tích viên và phát triển hệ thống cần phải tuân thủ và giữ vững trong quá trình phá triển phần mềm
  - Các đặc tính chất lượng (quality attributes): các đặc điểm xác định tính chất và chất lượng của phần mềm. Lưu ý cần nên rõ phương pháp đo và có sự thông nhất giữa NSD và các PTV về các đặc tính chất lượng này

Chương 1. Tổng quan về YCPM và quy trình TKXDPM

I. Các khái niệm căn bản và định nghĩa

1. Các khái niệm và định nghĩa
2. Quá trình phát triển của phương pháp tiếp cận thiết kế và xây dựng phần mềm
3. Phân loại phần mềm theo ứng dụng
4. Quy trình phát triển phần mềm (SDLC)
5. Một số lưu ý liên quan đến YCPM

II. Phân loại các yêu cầu phần mềm

III. Các thuộc tính chất lượng của yêu cầu phần mềm

IV. Quy trình yêu cầu phần mềm: mô hình, tác nhân, hỗ trợ và quản lí, kết quả

V.Câu hỏi và bài tập

VI. Giới thiệu một số tài liệu liên quan đến nội dung chương

VII. Nội dung đọc trước cho tuần sau

III. Các thuộc tính chất lượng của yêu cầu phần mềm

- Các đặc điểm của từng yêu cầu phần mềm:
  - (1) Hoàn thiện (complete)
  - (2) Chính xác (correct)
  - (3) Khả thi (feasible)
  - (4) Cần thiết (necessary)
  - (5) Được sắp xếp theo thứ tự ưu tiên các yêu cầu.
  - (6) Rõ ràng
  - (7) Kiểm tra được, xác minh được

III. Các thuộc tính chất lượng của yêu cầu phần mềm

- Các đặc điểm của tài liệu đặc tả yêu cầu phần mềm:
  - (1) Hoàn thiện (complete)
  - (2) Phù hợp (consistent)
  - (3) Sửa đổi được (modifiable)
  - (4) Theo dõi được (traceable)

Chương 1. Tổng quan về YCPM và quy trình TKXDPM

I. Các khái niệm căn bản và định nghĩa

1. Các khái niệm và định nghĩa
2. Quá trình phát triển của phương pháp tiếp cận thiết kế và xây dựng phần mềm
3. Phân loại phần mềm theo ứng dụng
4. Quy trình phát triển phần mềm (SDLC)
5. Một số lưu ý liên quan đến YCPM

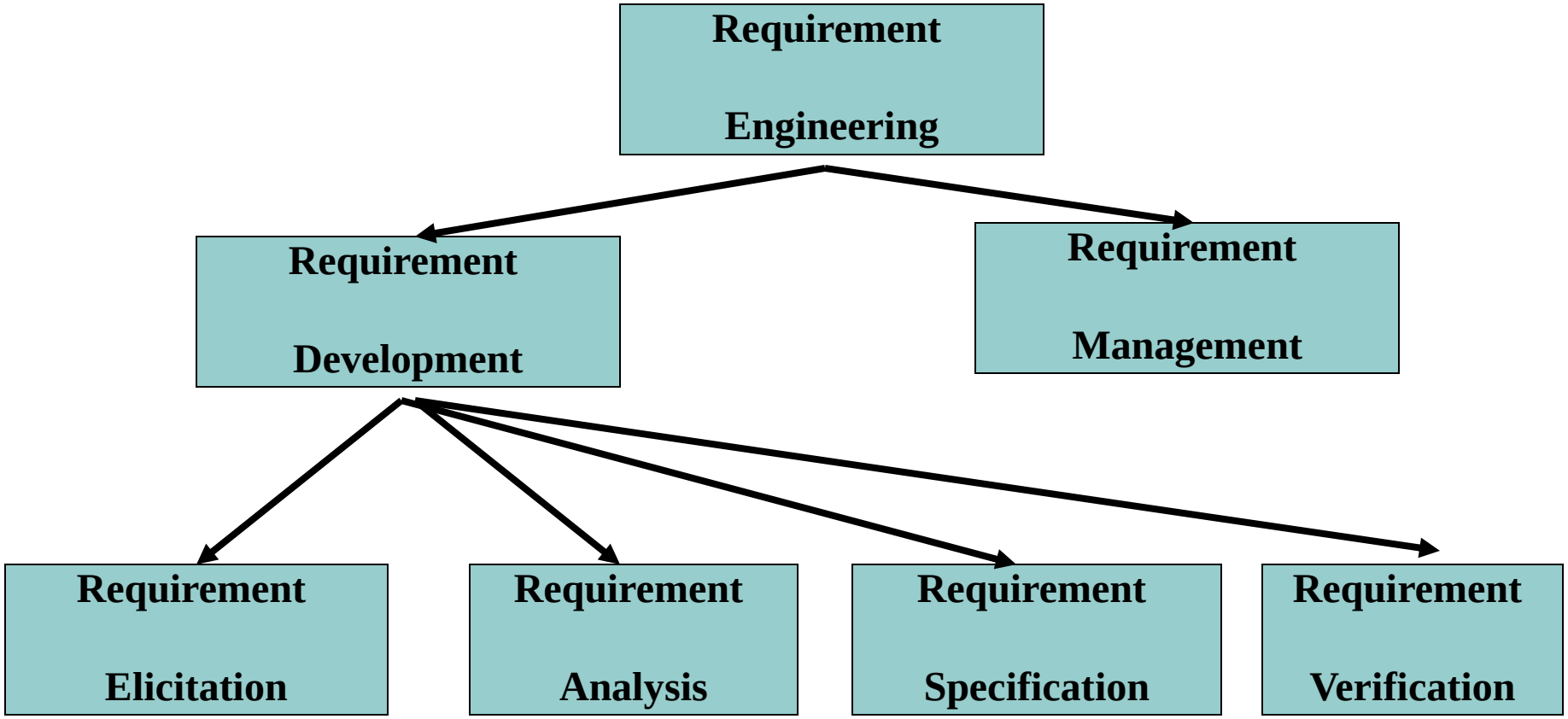
II. Phân loại các yêu cầu phần mềm

III. Các thuộc tính chất lượng của yêu cầu phần mềm

IV. Quy trình yêu cầu phần mềm: mô hình, tác nhân, hỗ trợ và quản lí, kết quả

V. Giới thiệu một số tài liệu liên quan đến nội dung chương

IV. Quy trình yêu cầu phần mềm: mô hình, tác nhân, hỗ trợ và quản lí, kết quả

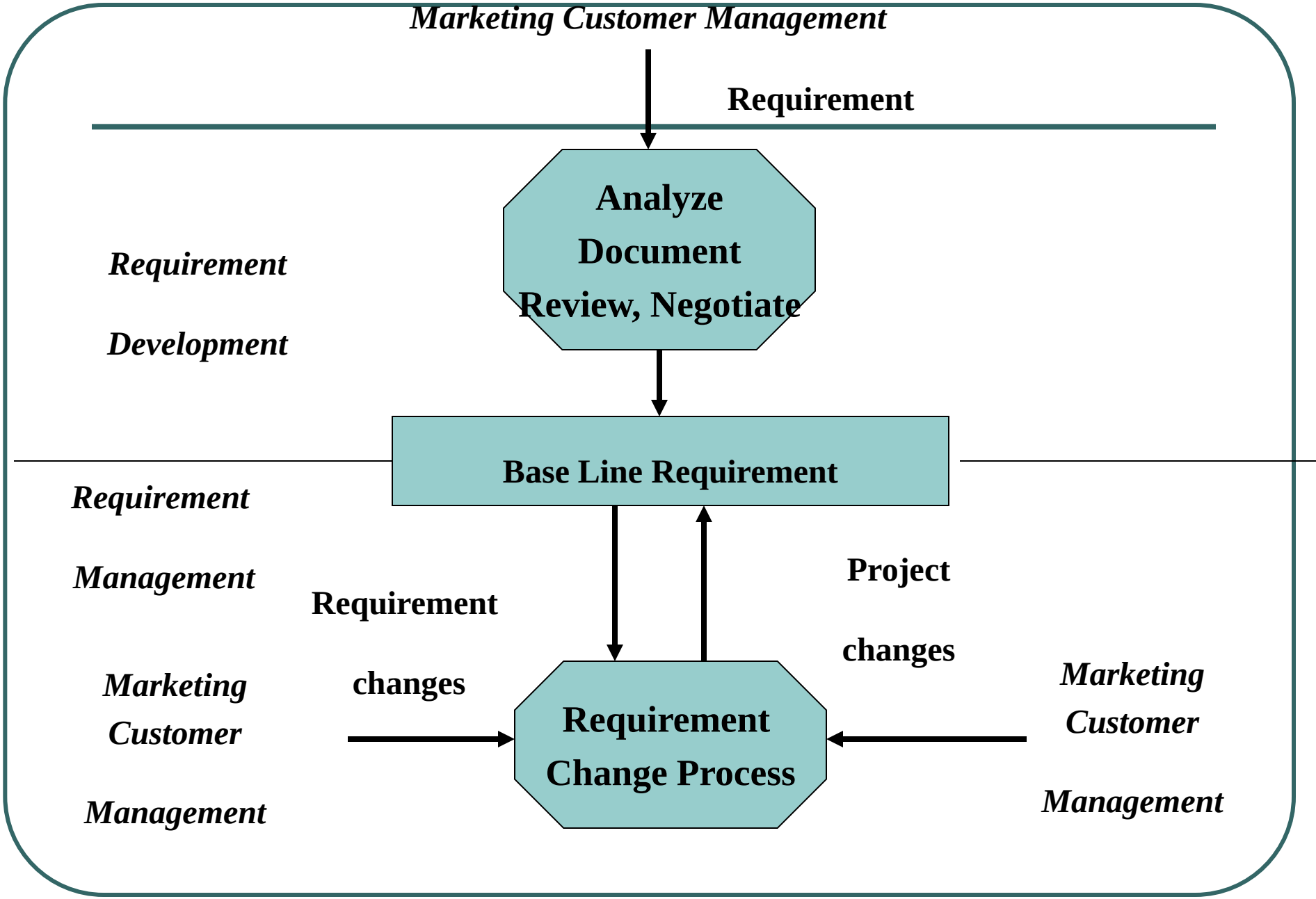


IV. Quy trình yêu cầu phần mềm: mô hình, tác nhân, hỗ trợ và quản lí, kết quả

- Requirement Development: (Phát triển các yêu cầu phần mềm
  - Phát hiện các yêu cầu (Requirement Elicitation)
  - Phân tích các yêu cầu (Requirement Analysis)
  - Đặc tả các yêu cầu (Requirement Specification)
  - Kiểm thử các yêu cầu (Requirement Verification)

IV. Quy trình yêu cầu phần mềm: mô hình, tác nhân, hỗ trợ và quản lí, kết quả

- Requirement Management: Quản lý các yêu cầu phần mềm thực hiện các giao tiếp và thoả thuận với NSD về các yêu cầu của phần mềm cần thực hiện (CMU/SEI 1995)
  - Xác định giới hạn của phần mềm (Requirement baseline)
  - Duyệt lại các giới hạn của phần mềm
  - Quản lý các thay đổi trong yêu cầu phần mềm (Requirement Changes)


Marketing Customer Management
Requirement
Analyze
Document
Review, Negotiate
Requirement
Development
Base Line Requirement
Requirement
Management
Requirement
changes
Project
changes
Marketing
Customer
Management
Requirement
Change Process
Marketing
Customer
Management

V. Giới thiệu một số tài liệu liên quan đến nội dung chương

1. Hoffer J. A. Modern System Analysis and Design. Third Edition. Addison-Wesley, 2004
2. Sommerville, I.: Software Engineering, 7th ed. Reading, Massachusetts: Addison-Wesley, 2005.
3. Guide to the Software Engineering Body of Knowledge 2004 Version. SWEBOK. IEEE project.
4. Địa chỉ trang Web: http://www.segvn.org/forum