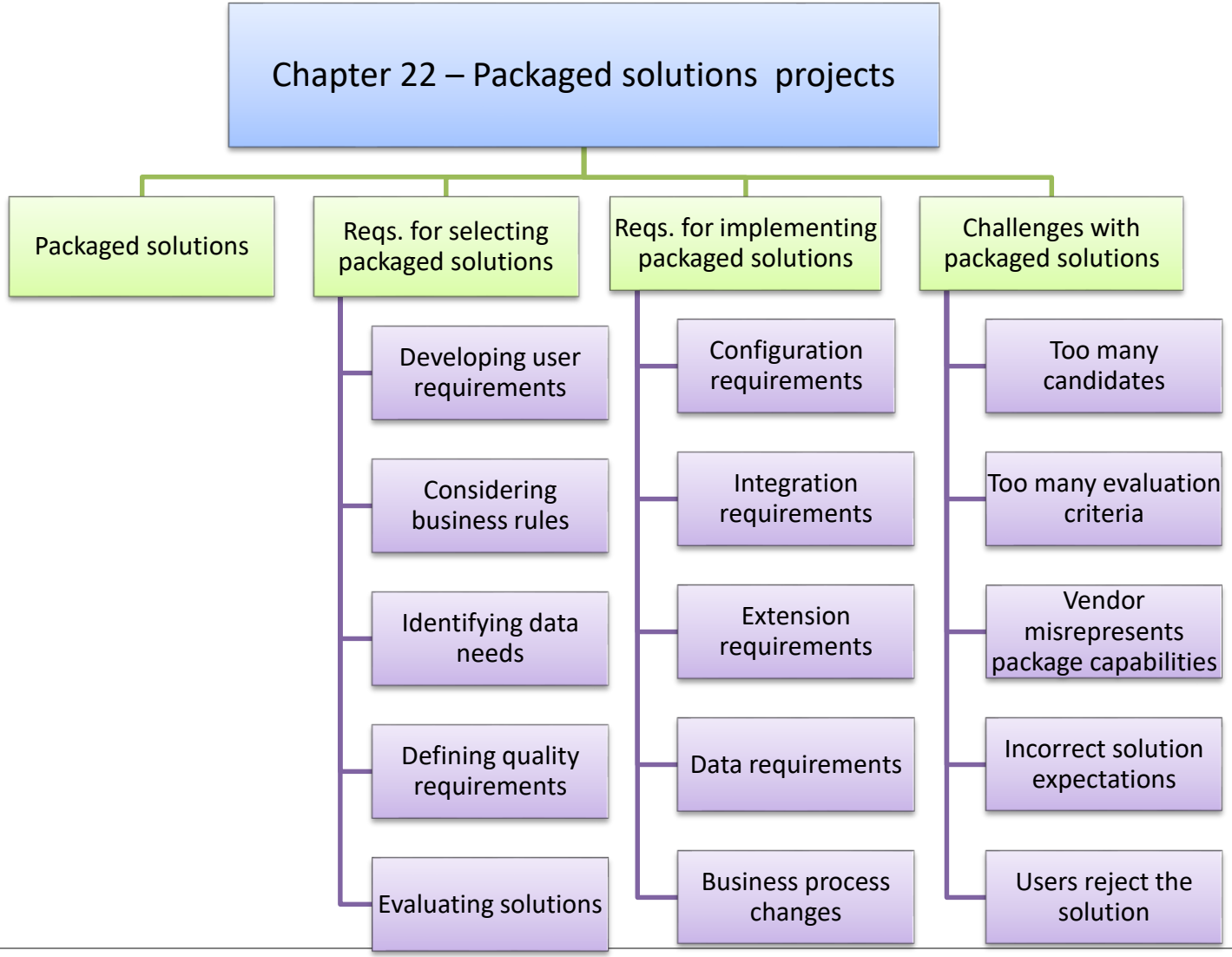
FPT UNIVERSITY

ĐẠI HỌC FPT CẦN THƠ

Chapter 27

Requirements management practices

Review chapter 22



Objectives

- Addresses some principles and practices of requirements management.
- Student should enhance that effective management and communication among the project participants take an important role in having great requirements.
- Student should only understand basic concepts. The more detail of every process will be clarified in the next chapters.

Contents

- Requirements management process
- Requirement attributes
- Tracking requirements status
- Resolving requirements issues
- Common types of requirements issues
- Measuring requirements effort
- Managing requirements on agile projects

Requirements management process

- Requirements management includes all activities that maintain the integrity, accuracy, and currency (Sự phổ biến) of requirements agreements throughout the project.

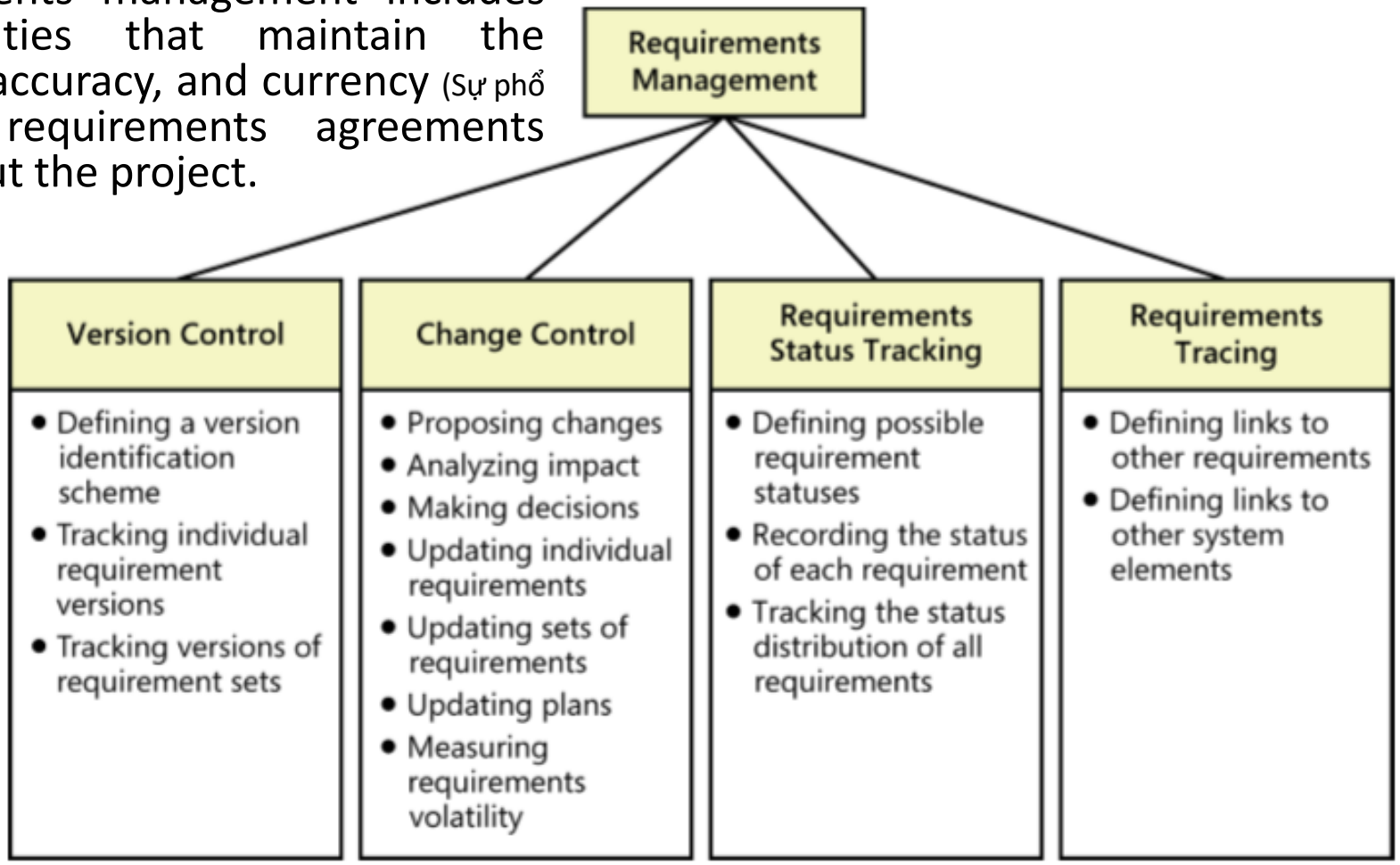


FIGURE 27-1 Major requirements management activities.

Requirements baseline

- Requirements development involves activities to elicit, analyze, specify, and validate a software project's requirements.
- Requirements development deliverables include business requirements, user requirements, functional and nonfunctional requirements, a data dictionary, and various analysis models.
- After they are reviewed and approved, any defined subset of these items constitutes a *requirements baseline*.
- *A requirements baseline* is a set of requirements that stakeholders have agreed to, often defining the contents of a specific planned release or development iteration.
- Subsequent changes can *be made only* through the project's defined change control procedure.

Requirements version control

- Version control—uniquely identifying different versions of an item—applies at the level of both individual requirements and requirements sets, most commonly represented in the form of documents.
- Begin version control as soon as you draft a requirement or a document so you can retain a history of changes made.
- Every version of the requirements must be uniquely identified. Every team member must be able to access the current version of the requirements.
- Changes must be clearly documented and communicated (truyền đạt) to everyone affected.
- To minimize confusion and miscommunication, permit only designated individuals to update the requirements, and make sure that the version identifier changes whenever an update is made.
- Each circulated (lưu hành) version of a requirements document should include a revision history that identifies the changes made, the date of each change, the individual who made the change, and the reason for each change.

Requirement attributes

- Think of each requirement as an object with properties that distinguish it from other requirements.
- In addition to (ngoài) its textual description, each requirement should have supporting pieces of information or *attributes* associated with it. These attributes establish a context and background for each requirement.
- Following is a list of potential requirement attributes to consider:
  - Date the requirement was created.
  - Current version number of the requirement.
  - Author who wrote the requirement.
  - Priority.
  - Status.
  - Origin or source of the requirement.
  - Rationale behind the requirement.
  - Release number or iteration to which the requirement is allocated.
  - Stakeholder to contact with questions or to make decisions about proposed changes.
  - Validation method to be used or acceptance criteria.

Tracking requirements status

- Like nearly everyone, software developers are sometimes overly optimistic when they report how much of a task is complete.
- Tracking status means comparing where you really are at a particular time against (so với) the expectation of what "complete" means for this development cycle.
- Next slide shows you the suggested requirement statuses.

Tracking requirements status

TABLE 27-1 Suggested requirement statuses

Status	Definition
Proposed	The requirement has been requested by an authorized source.
In Progress	A business analyst is actively working on crafting the requirement.
Drafted	The initial version of the requirement has been written.
Approved	The requirement has been analyzed, its impact on the project has been estimated, and it has been allocated to the baseline for a specific release. The key stakeholders have agreed to incorporate the requirement, and the software development group has committed to implement it.
Implemented	The code that implements the requirement has been designed, written, and unit tested. The requirement has been traced to the pertinent design and code elements. The software that implemented the requirement is now ready for testing, review, or other verification.
Verified	The requirement has satisfied its acceptance criteria, meaning that the correct functioning of the implemented requirement has been confirmed. The requirement has been traced to pertinent tests. It is now considered complete.
Deferred	An approved requirement is now planned for implementation in a later release.
Deleted	An approved requirement has been removed from the baseline. Include an explanation of why and by whom the decision was made to delete it.
Rejected	The requirement was proposed but was never approved and is not planned for implementation in any upcoming release. Include an explanation of why and by whom the decision was made to reject it.

Resolving requirements issues

- Numerous questions, decisions, and issues related to requirements will arise during the course of a project.
- Potential issues include items flagged as TBD, pending decisions, information that is needed, and conflicts awaiting resolution. Issues from multiple requirements reviews are collected so that no issue ever gets lost.
- Some of the benefits of using an issue-tracking tool are:
  - The project manager can easily see the current status of all issues.
  - A single owner can be assigned to each issue.
  - The history of discussion around an issue can be retained.
  - The team can begin development earlier with a known set of open issues rather than having to wait until the SRS is complete.

Common types of requirements issues

- Nearly all of the defects logged early in a project are related to issues in the requirements, such as asking for clarification on a requirement, scope decisions, questions about development feasibility, and to-do items on the requirements themselves.
- All stakeholders can log questions as they review the requirements. Table 27-2 lists several common types of requirements issues that can arise.

TABLE 27-2 Common types of requirements issues

Issue type	Description
Requirement question	Something isn't understood or decided about a requirement.
Missing requirement	Developers uncovered a missed requirement during design or implementation.
Incorrect requirement	A requirement was wrong. It should be corrected or removed.
Implementation question	As developers implement requirements, they have questions about how something should work or about design alternatives.
Duplicate requirement	Two or more equivalent requirements are discovered. Delete all but one of them.
Unneeded requirement	A requirement simply isn't needed anymore.

Measuring requirements effort

- As with requirements development, your project plan should include tasks and resources for the requirements management activities described in this chapter.
- Record the number of hours spent on *requirements development activities* such as the following:
  - Planning requirements-related activities for the project.
  - Holding workshops and interviews, analyzing documents, and performing other elicitation activities.
  - Writing requirements specifications, creating analysis models, and prioritizing requirements.
  - Creating and evaluating prototypes intended to assist with requirements development.
  - Reviewing requirements and performing other validation activities.

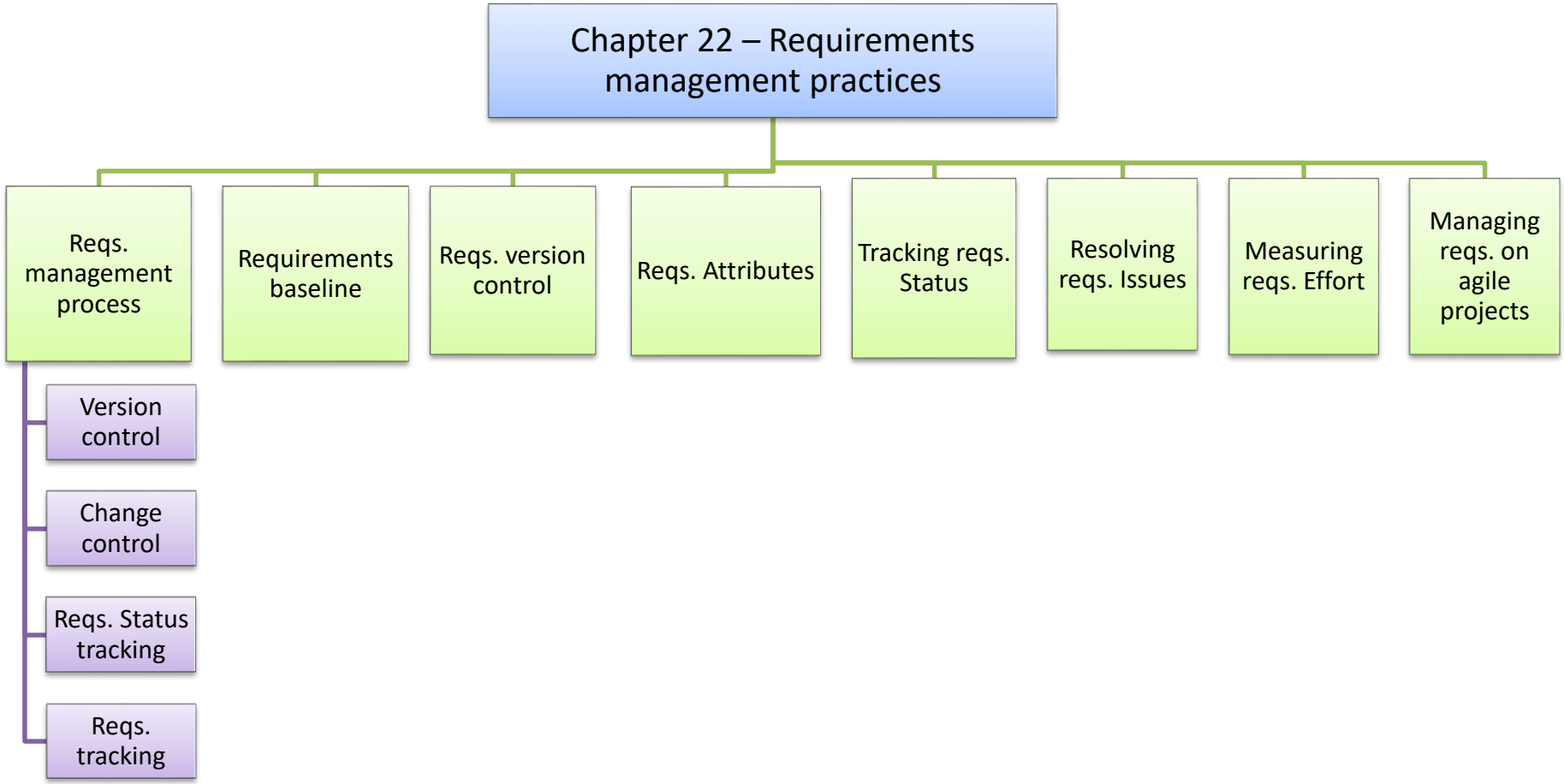
Measuring requirements effort

- Count the effort devoted to the following activities as *requirements management* effort:
  - Configuring a requirements management tool for your project.
  - Submitting requirements changes and proposing new requirements.
  - Evaluating proposed changes, including performing impact analysis and making decisions.
  - Updating the requirements repository.
  - Communicating requirements changes to affected stakeholders.
  - Tracking and reporting requirements status.
  - Creating requirements trace information.

Managing requirements on agile projects

- Agile projects accommodate (điều tiết) change by building the product through a series of development iterations and managing a dynamic product backlog of work remaining to be done.
- Some agile teams, particularly large or distributed teams, use an agile project management tool to track the status of an iteration and the stories allocated to it. The stories and their associated acceptance criteria and acceptance tests might all be placed in a product backlog or user story–management tool.
- Story status can be monitored by using statuses:
  - In backlog (the story is not yet allocated to an iteration)
  - Defined (details of the story were discussed and understood, and acceptance tests were written)
  - In progress (the story is being implemented)
  - Completed (the story is fully implemented)
  - Accepted (acceptance tests were passed)
  - Blocked (the developer is unable to proceed until something else is resolved)

Review chapter 27



THE END
THANK YOU!