


Recall The Team Skills

1. Analyzing the Problem (with 5 steps)
2. Understanding User and Stakeholder Needs
3. **Defining the System**
 - A Use Case Primer
 - Organizing Requirements Information
 - The Vision Document
4. Managing Scope
5. Refining the System Definition
6. Building the Right System

Chapter 15

Organizing Requirements Information



- why?
- Organizing req.s for complex systems
- Organizing req.s for family of systems

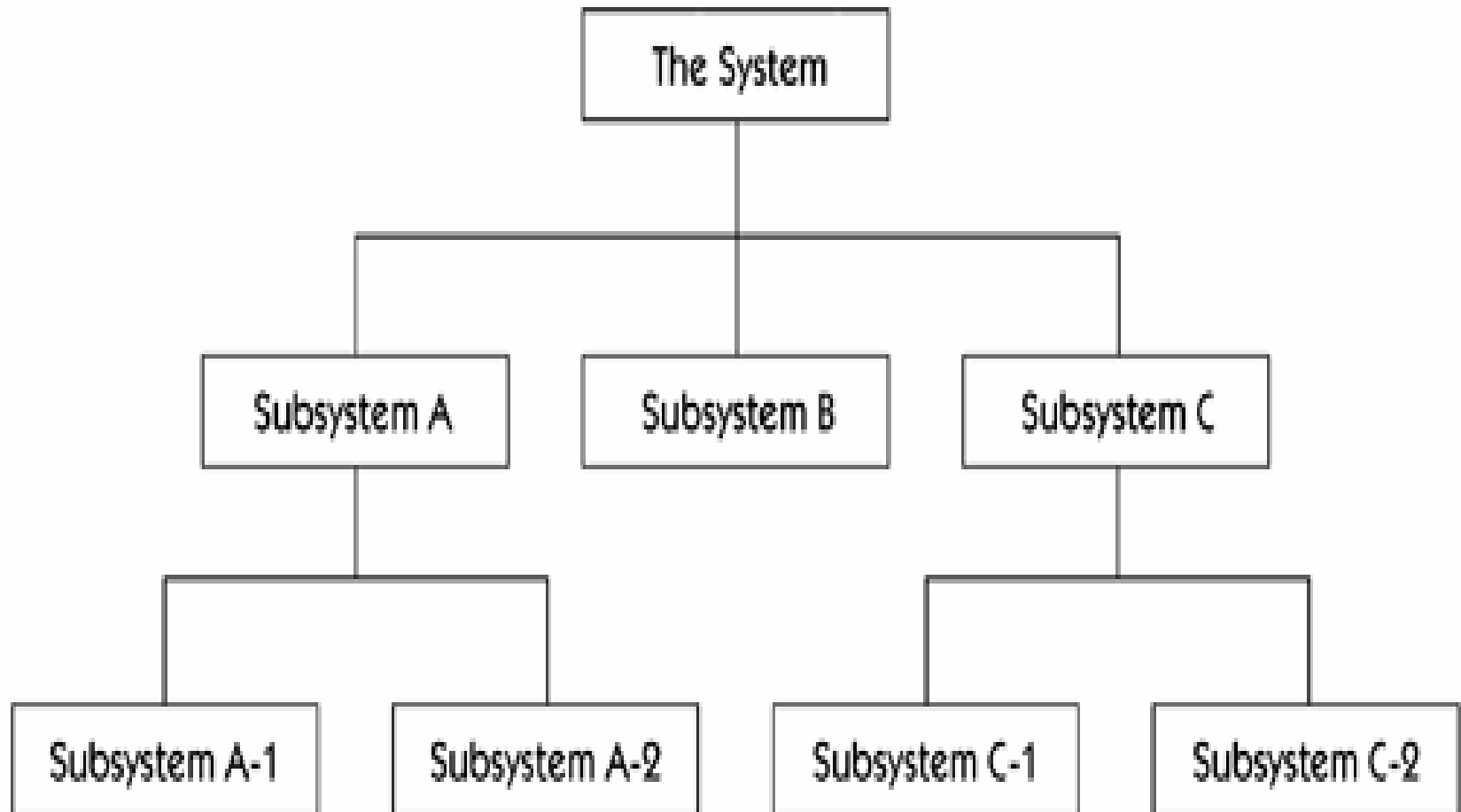
Why ..?

- ❑ Communication problems demand that requirements should be captured and documented in a way that they can be reviewed, approved, agreed on and referred to easily.
- ❑ Traditionally this is done by requirements specifications which are large documents built to capture and communicate the external behaviour of the system.

Why ..?

- Requirements can rarely be defined in a single document or in a single use-case model because the system may
 - be very complex
 - be a member of a family of related products.
 - be a subsystem of a larger system and may satisfy only a subset of all the requirements identified.
 - contain marketing and business goals needed to be separated from the detailed product requirements.
 - have other requirements, perhaps regulatory or legal, which may be documented elsewhere.

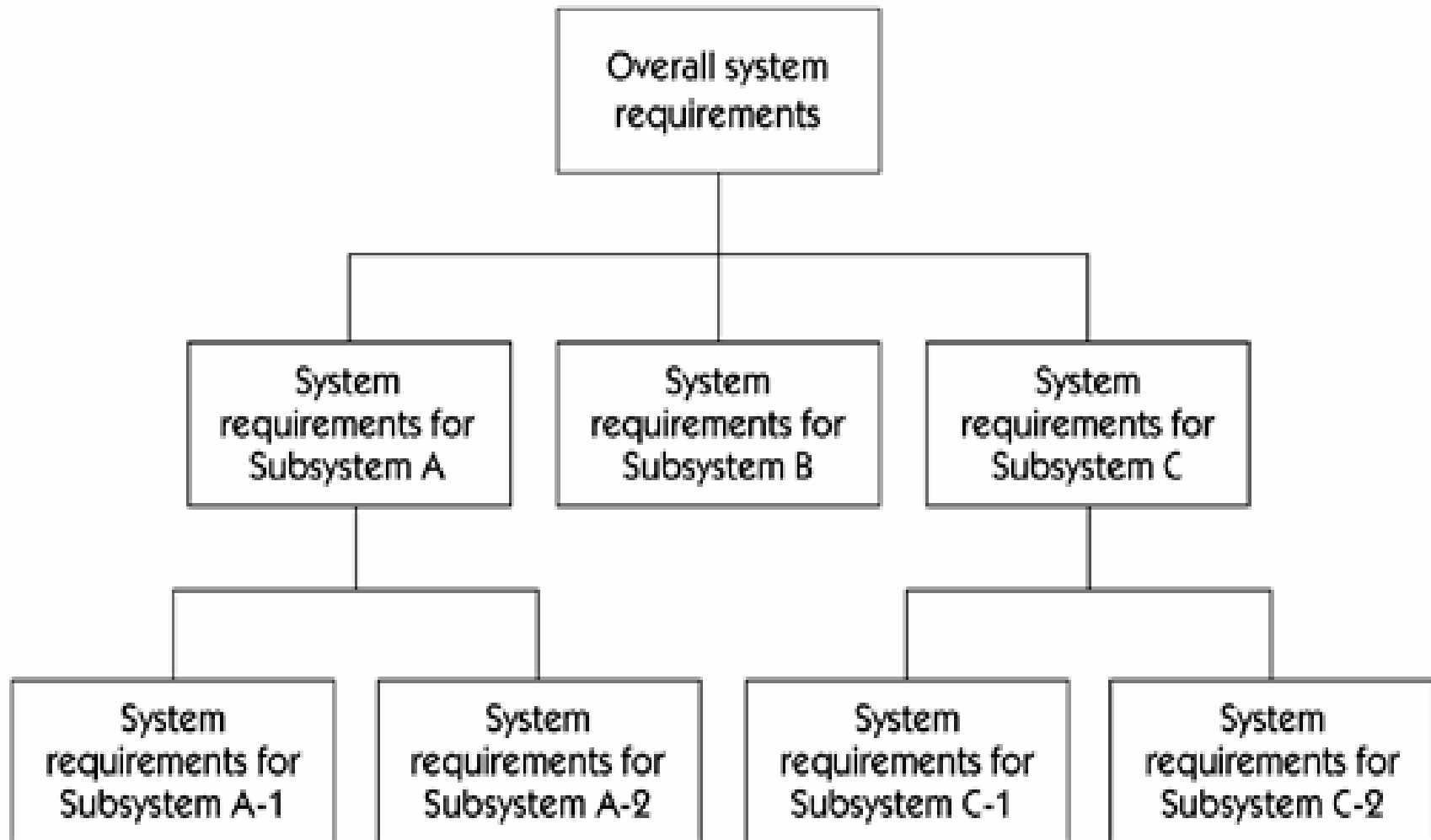
Complex system: A System of Subsystems



Organizing Requirements of Complex Systems

- Create a system-level requirements set that
 - describes the system and
 - the system-level use cases which describe functional behaviour without knowledge of (or reference to) any of its subsystems.
- Next, develop requirements for each subsystem that
 - describe the external behaviour of the subsystem completely, without reference to any of its subsystems (sub-subsystems). And so on ...

Hierarchy of requirements resulting from system design



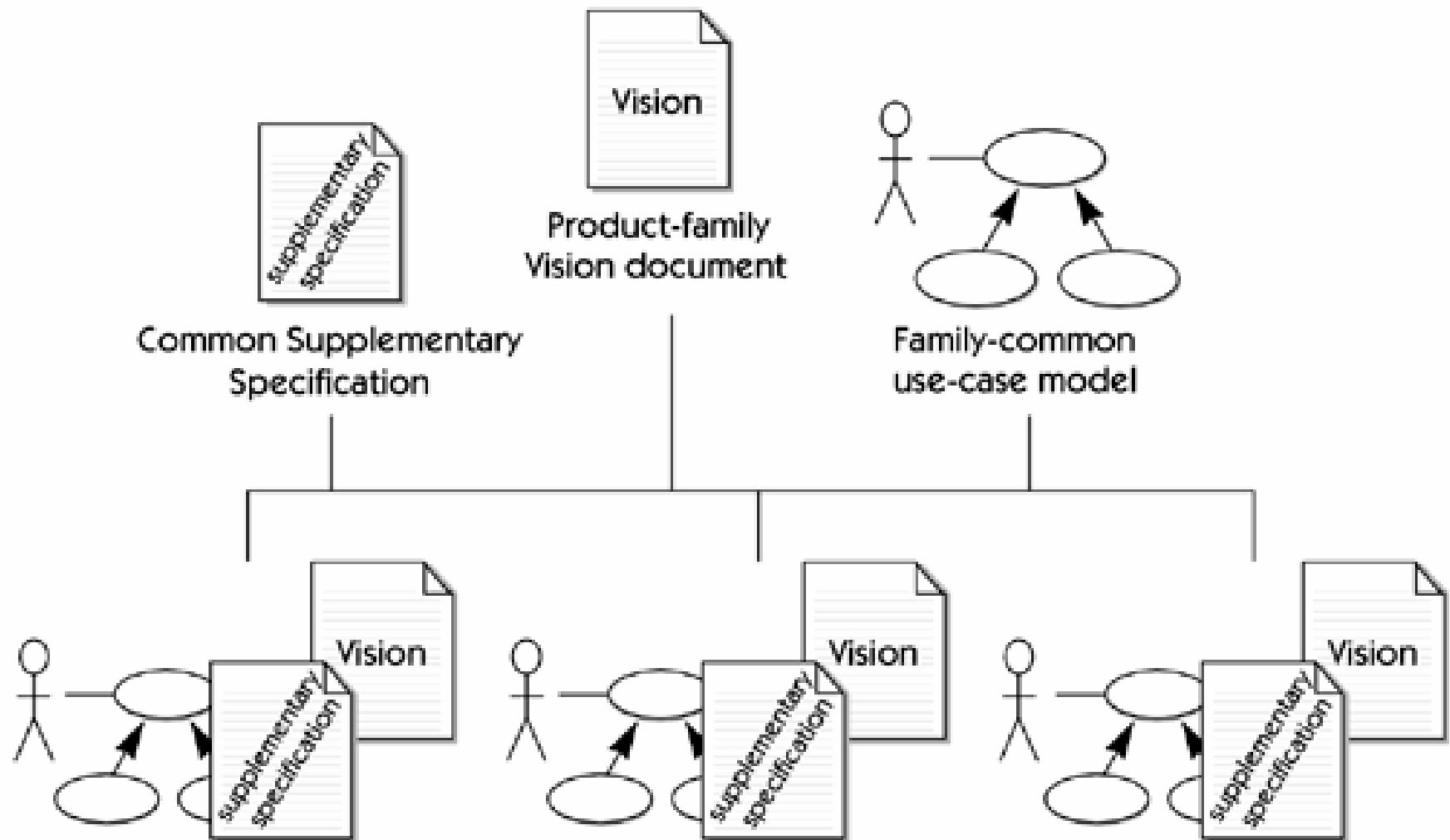
Organizing Requirements for Product Families

- ❑ Many industries build sets of closely related products that have much functionality in common, yet each product contains some unique features.
- ❑ **Examples:** inventory control systems, telephone answering machines, application development tools.

Organizing Requirements for Product Families: Approach

- ❑ Develop a product-family Vision document that describes the ways in which the products are intended to work together and the other features that could be shared.
- ❑ To better understand the shared-usage model, you might also develop a set of use cases showing how the users will interact with various applications running together.
- ❑ Develop a common software requirements set that defines the specific requirements for shared functionality, such as menu structures, common GUIs, and communication protocols.
- ❑ For each product in the family, develop a Vision document, supplementary specification, and a use-case model that defines its specific functionality.

Requirements Organization for a Software Product Family



On "Future" Requirements

- During any process of requirements elicitation, requirements will arise that are deemed appropriate for the next release of the product.
- It makes sense to record both current and future requirements but to clearly identify those requirements that are planned for the current release. Reasons:
 - Future requirements represent value-added work products.
 - We want to produce requirements from them for future releases.
 - The system designers may well have designed the system differently had they known that future requirements of a certain type were desired.

Reading Assignments

- Read the HOLIS case study in page 171.

Key Points

- ❑ For nontrivial applications, requirements must be captured and recorded in a document, database, model, or tool.
- ❑ Different types of projects require different requirements organization techniques.
- ❑ Complex systems require that requirements sets be developed for each subsystem.