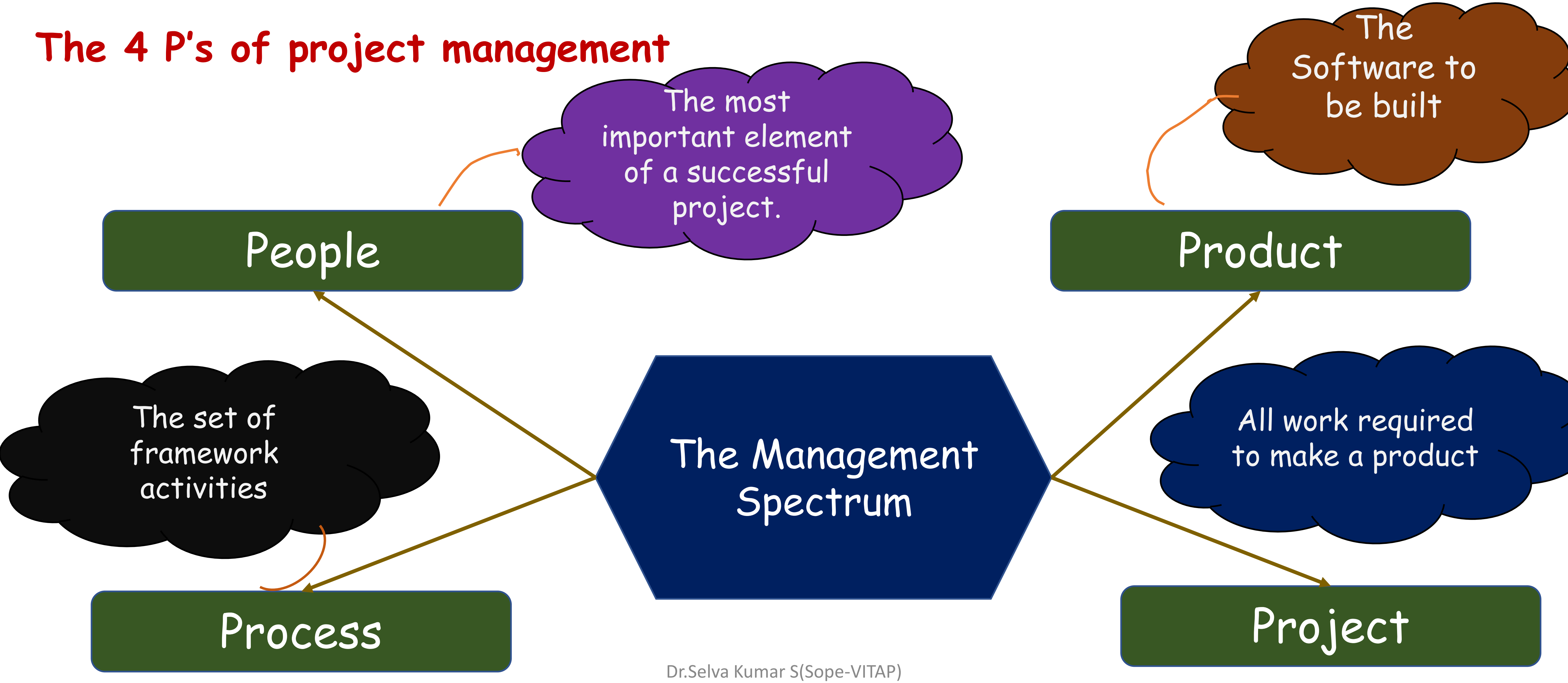
Module 5

Managing Software Projects

The Management Spectrum

The 4 P's of project management



The most important element of a successful project.
The Software to be built
People
Product
The set of framework activities
The Management Spectrum
All work required to make a product
Process
Project

The Management Spectrum

1. People

- This is virtually any type of stakeholder who is involved in the project. The most important element of a successful project.
- People of a project include from manager to developer, from client to End user. however principally people of a project highlight the developers.

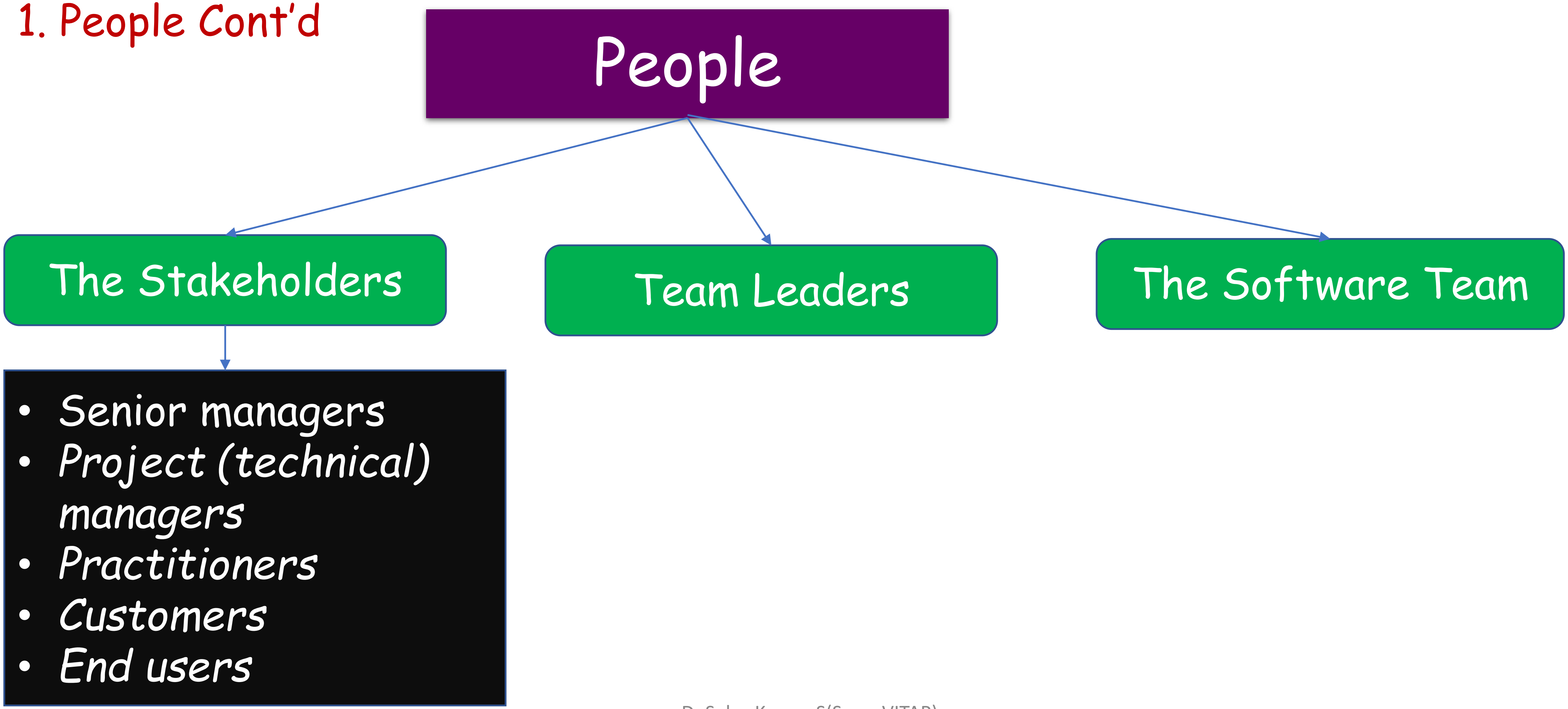
People Capability Maturity Model (People-CMM),

- Staffing,
- Communication and coordination,
- Work environment,
- Performance management,
- Training,
- Compensation,

- Competency analysis and development,
- Career development,
- Workgroup development,
- Team/culture development,
- and others..,

The Management Spectrum

1. People Cont'd



People

The Stakeholders

Team Leaders

The Software Team

- Senior managers
- *Project (technical) managers*
- *Practitioners*
- *Customers*
- *End users*

The Management Spectrum

1. People Cont'd

Stakeholders

- *Senior managers* who define the business issues that often have a significant influence on the project.
- *Project (technical) managers* who must plan, motivate, organize, and control the practitioners who do software work.
- *Practitioners* who deliver the technical skills that are necessary to engineer a product or application.
- *Customers* who specify the requirements for the software to be engineered and other stakeholders who have a peripheral interest in the outcome.
- *End-users* who interact with the software once it is released for production use.

The Management Spectrum

1. People Cont'd

Team Leader

- A Software Engineering Team Leader is responsible for their team's execution, the quality they produce, the speed and cadence at which they produce, but most importantly, they are responsible for the team's culture, environment, and overall growth of the people on it.

The MOI Model

Motivation. The ability to encourage (by "push or pull") technical people to produce to their best ability.

Organization. The ability to mold existing processes (or invent new ones) that will enable the initial concept to be translated into a final product.

Ideas or innovation. The ability to encourage people to create and feel creative even when they must work within bounds established for a particular software product or application.

The Management Spectrum

1. People Cont'd

Project manager

- Software project managers are in charge of the planning, scheduling, budgeting, execution, and delivery of software and web projects.

The characteristics that define an effective project manager emphasizes four key traits:

- Problem solving
- Managerial identity
- Achievement
- Influence and team building

The Management Spectrum

1. People Cont'd

The Software Team

The seven project factors should be considered when planning the structure of software engineering teams

- The difficulty of the problem to be solved
- The size of the resultant program(s) in lines of code or function points
- The time that the team will stay together (team lifetime)
- The degree to which the problem can be modularized
- The required quality and reliability of the system to be built
- The rigidity of the delivery date
- The degree of sociability (communication) required for the project

The Management Spectrum

1. People Cont'd

The Software Team Cont'd

Organizational paradigms *suggested by Constantine [Con93]*

- Closed paradigm—structures a team along a traditional hierarchy of authority
- Random paradigm—structures a team loosely and depends on the individual initiative of the team members
- Open paradigm—attempts to structure a team in a manner that achieves some of the controls associated with the closed paradigm but also much of the innovation that occurs when using the random paradigm
- Synchronous paradigm—relies on the natural compartmentalization of a problem and organizes team members to work on pieces of the problem with little active communication among themselves

The Management Spectrum

1. People Cont'd

The Software Team Cont'd

To achieve a high-performance team

- Team members must have trust in one another.
- The distribution of skills must be appropriate to the problem.
- Mavericks may have to be excluded from the team, if team cohesiveness is to be maintained.

Avoid Team "Toxicity"

(1) a frenzied work atmosphere,

(2) high frustration that causes friction among team members,

(3) a "fragmented or poorly coordinated" software process,

(4) an unclear definition of roles on the software team, and

(5) "continuous and repeated exposure to failure."

The Management Spectrum

1. People Cont'd

Agile Teams

YOU
STAKEHOLDERS
SCRUM MASTER
PRODUCT OWNER
SCRUM TEAM
DEVELOPMENT TEAM

- The small, highly motivated project team, also called an *agile team*,

Team is "self-organizing"

- An adaptive team structure
- Uses elements of Constantine's random, open, and synchronous paradigms
- Significant autonomy

The Management Spectrum

1. People Cont’d

Team Coordination & Communication

- *Formal, impersonal approaches* include software engineering documents and work products (including source code), technical memos, project milestones, schedules, and project control tools, change requests and related documentation, error tracking reports, and repository data.

- *Formal, interpersonal procedures* focus on quality assurance activities applied to software engineering work products.
- *Informal, interpersonal procedures* include group meetings for information dissemination and problem solving and “collocation of requirements and development staff.”

- *Electronic communication* encompasses electronic mail, electronic bulletin boards, and by extension, video-based conferencing systems.

- *Interpersonal networking* includes informal discussions with team members and those outside the project who may have experience or insight that can assist team members.

The Management Spectrum

2. The Product

- The software product is software that has been developed and maintained for the benefit of a user base and often to satisfy a need in the market.

2.1 Software Scope

The first software project management activity is the determination of *software scope*.

Scope is defined by answering the following questions

- Context. How does the software to be built fit into a larger system, product, or business context, and what constraints are imposed as a result of the context?
- Information objectives. What customer-visible data objects are produced as output from the software? What data objects are required for input?
- Function and performance. What function does the software perform to transform input data into output? Are any special performance characteristics to be addressed?

The Management Spectrum

2. The Product Cont'd

2.2 Problem Decomposition

- Problem decomposition, sometimes called *partitioning or problem elaboration*, is an activity that sits at the core of software requirements analysis.
- It involves breaking down a complex problem or system into smaller parts that are more manageable and easier to understand.
- The smaller parts can then be examined and solved, or designed individually, as they are simpler to work with.

The decomposition process continues until all functions or problem classes have been defined.

If a problem is not decomposed, it is much harder to solve

The Management Spectrum

3. The Process

- The framework activities that characterize the software process are applicable to all software projects.
- The problem is to select the process model that is appropriate for the software to be engineered.

The team must decide which process model is most appropriate for

(1) The customers who have requested the product and the people who will do the work,

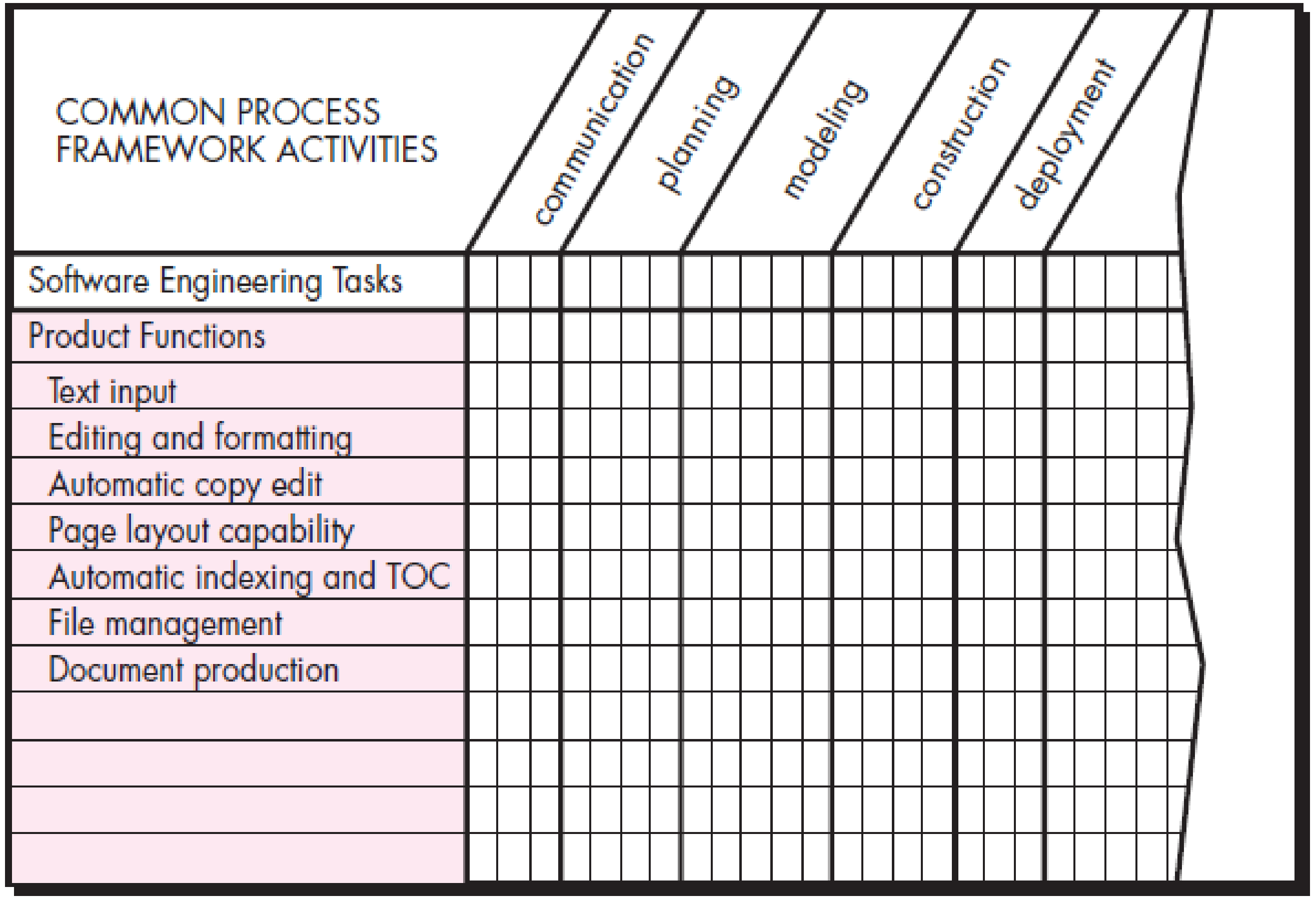
(2) The characteristics of the product itself, and

(3) The project environment in which the software team works.

The Management Spectrum

3. The Process Cont'd

Melding the Product and the Process



COMMON PROCESS FRAMEWORK ACTIVITIES	communication	planning	modeling	construction	deployment
Software Engineering Tasks					
Product Functions					
Text input					
Editing and formatting					
Automatic copy edit					
Page layout capability					
Automatic indexing and TOC					
File management					
Document production					

The generic framework activities:

- Communication,
- Planning,
- Modeling,
- Construction,
- and Deployment

- Each major product function is listed in the left-hand column. Framework activities are listed in the top row.

The Management Spectrum

4. The Project

- A project is a well-defined task, which is a collection of several operations done in order to achieve a goal (for example, software development and delivery).

Projects get into trouble when (John Reel,1999)

1. Software people don't understand their customer's needs.
2. The product scope is poorly defined.
3. Changes are managed poorly.
4. The chosen technology changes.
5. Business needs change [or are ill-defined].
6. Deadlines are unrealistic.
7. Users are resistant.
8. Sponsorship is lost [or was never properly obtained].
9. The project team lacks people with appropriate skills.
10. Managers [and practitioners] avoid best practices and lessons learned.

The Management Spectrum

4. The Project Cont'd

Common-sense Approach To Projects

- *Start on the right foot.* This is accomplished by working hard (very hard) to understand the problem that is to be solved and then setting realistic objectives and expectations.
- *Maintain momentum.* The project manager must provide incentives to keep turnover of personnel to an absolute minimum, the team should emphasize quality in every task it performs, and senior management should do everything possible to stay out of the team's way.
- *Track progress.* For a software project, progress is tracked as work products (e.g., models, source code, sets of test cases) are produced and approved (using formal technical reviews) as part of a quality assurance activity.
- *Make smart decisions.* In essence, the decisions of the project manager and the software team should be to "keep it simple."
- *Conduct a postmortem analysis.* Establish a consistent mechanism for extracting lessons learned for each project.

The Management Spectrum

To Get to the Essence of a Project

- Why is the system being developed?
- What will be done?
- When will it be accomplished?
- Who is responsible?
- Where are they organizationally located?
- How will the job be done technically and managerially?
- How much of each resource (e.g., people, software, tools, database) will be needed?

The Management Spectrum

Critical Practices

- Formal risk management
- Empirical cost and schedule estimation
- Metrics-based project management
- Earned value tracking
- Defect tracking against quality targets
- People aware project management