

소프트웨어 제품 품질평가를 위한 국제표준화



2004-04-17

이금석

동국대학교 컴퓨터멀티미디어공학과



Contents

- 1: **Introduction** to Software Product Quality, and ISO/IEC 9126 and 14598 Series
- 2: **Quality Model**
- 3: Quality Life Cycle and **Methodologies**
- 4: **Measurement** and Measures
- 5: Software Products Evaluation Planning and **Management**

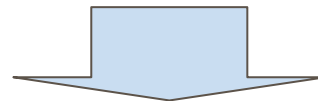
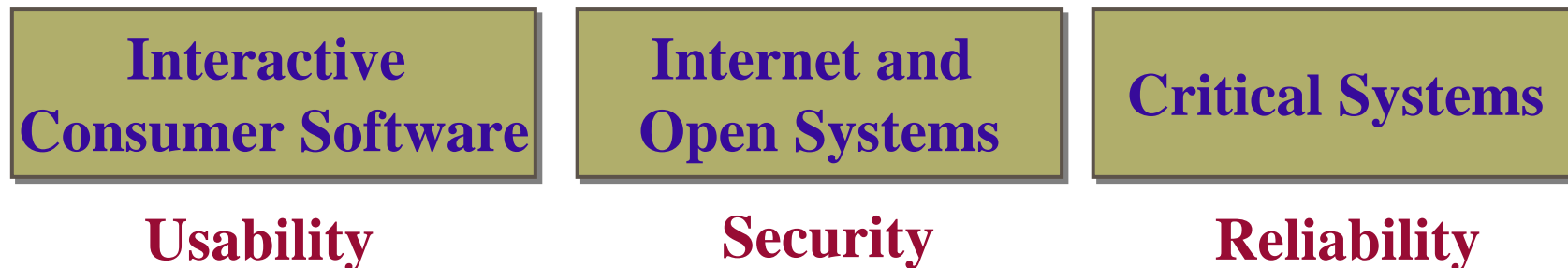
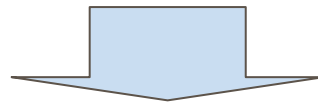


Part 1:
INTRODUCTION to
SOFTWARE PRODUCT QUALITY
ISO/IEC 9126 and 14598 SERIES



BACKGROUND

- Advance in Information Technologies
- High Performance / Price Hardware
- Rapid Growth of Internet Service and WWW



Software quality is crucially important!!



What does QUALITY Mean ?

- **Quality Definition (ISO 8402, 1994)**
 - The totality of characteristics of an entity that bear on its ability to satisfy stated and implied needs.
- **But what does “Needs” mean?**



NEEDS

- “Needs” is expectations for the effects of a product.
- A user wants not a product itself but the effects of the product, when it is used.
- A user does not know the real needs until he/she actually use it.
- User’s needs may change.
- It is difficult that real needs be identified either by a user or a product-planner.

CATEGORIES of SOFTWARE CRITICALITY

Criticality and Property

- **National Defense Critical**
 - Security, Reliability
- **Human Life Critical**
 - Correctness, Safety
- **Social Environment Critical**
 - Reliability, Security
- **Corporate Critical**
 - Effectiveness, Efficiency
- **Users' Health Critical**
 - Usability, Attractiveness

System Example

Defense System
Budget System

Medical System
Air Traffic Control System

Banking System
Telephone Switching System

Production System
Customer Database System

Interactive Systems
Computer Game



How to Improve Software Product Quality

- Develop better corporate culture, tools and environment, and supporting function.
- Define quality requirement for all quality characteristics using Measures.
- Select and assign resources appropriate to the requirements.
- Measure and evaluate product quality at every possible stage.
- Assess, control and improve process.



Why Measuring and Evaluating Software Products ?

■ Resources and Capability Quality

- In order to go the goal place fast and comfortable, a **good car and a skilled driver** are key to success.

■ Process Quality

- In order to go to the goal place efficiently, **good drive map and guides** are necessary.

■ Product Quality

- You cannot guide the way to the goal if you don't know **where you are**.



ISO/IEC JTC1/SC7/WG6

EVALUATION AND METRICS

■ **History**

- Original idea proposed in 1978 at Stockholm Meeting.
- Project initiated in 1985 at Stockholm Meeting.

■ **Output**

- **ISO/IEC 9126 , 1991** : Information technology - Software product evaluation - Quality characteristics and guidelines for their use
- **ISO/IEC 12119 , 1994** : Information technology - Software packages - Quality Requirement and testing



International Standards by SC7/WG6

ISO/IEC 9126: Software Product Quality

- **Part 1: Quality Model**
 - IS, Published, 2001
- **Part 2: External Metrics**
 - TR, To be published
- **Part 3: Internal Metrics**
 - TR , To be published
- **Part 4: Quality In Use Metrics**
 - TR , To be published



International Standards by SC7/WG6

ISO/IEC 14598: Software Product Evaluation

- Part 1: General Overview
- Part 2: Planning and Management
- Part 3: Process for Developers
- Part 4: Process for Acquirers
- Part 5: Process for Evaluators
- Part 6: Documentation of Evaluation Module

All parts are now available.



Part 2: QUALITY MODEL

ISO/IEC 9126-1



Definitions from ISO/IEC 14598-1

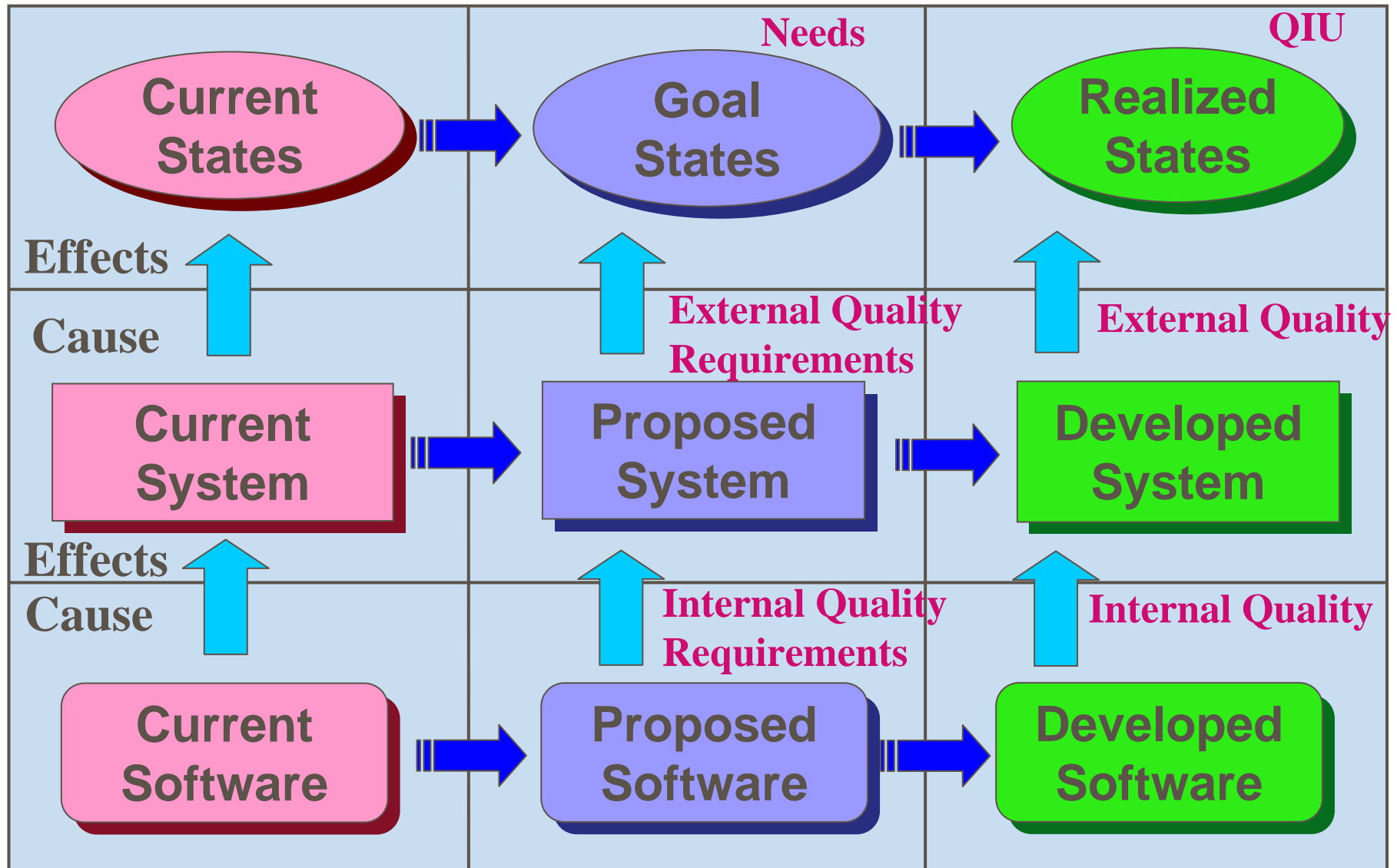
- **Attribute:** Measurable physical or abstract property of an entity.
- **Metric:** Defined measurement method and the measurement scale (**Measurement System**)
- **Measure:** Number or category assigned to an attribute of an entity by making a measurement (**Variable to which a value is assigned as the result of measurement**)
- **Measurement:** Set of operations having the object of determining a value of a measure



QUALITY MODEL

- **ISO/IEC 14598-1 requires it's audience to use a quality model**
- **Definition of the Quality Model (ISO/IEC 14598-1)**
 - The set of **characteristics and the relationships between** them which provides the basis for specifying requirements and evaluating quality.
- **Quality Model is the results of Quality Deployment.**
- **Early works**
 - Boehm Model (1976) and McCall Model (1977)
 - ISO/IEC 9126 (1991)

Relation of States - System – Software





EXTERNAL QUALITY DEFINITION

- **The extent to which a product satisfies stated and implied needs when used under specified conditions (ISO/IEC 14598-1)**

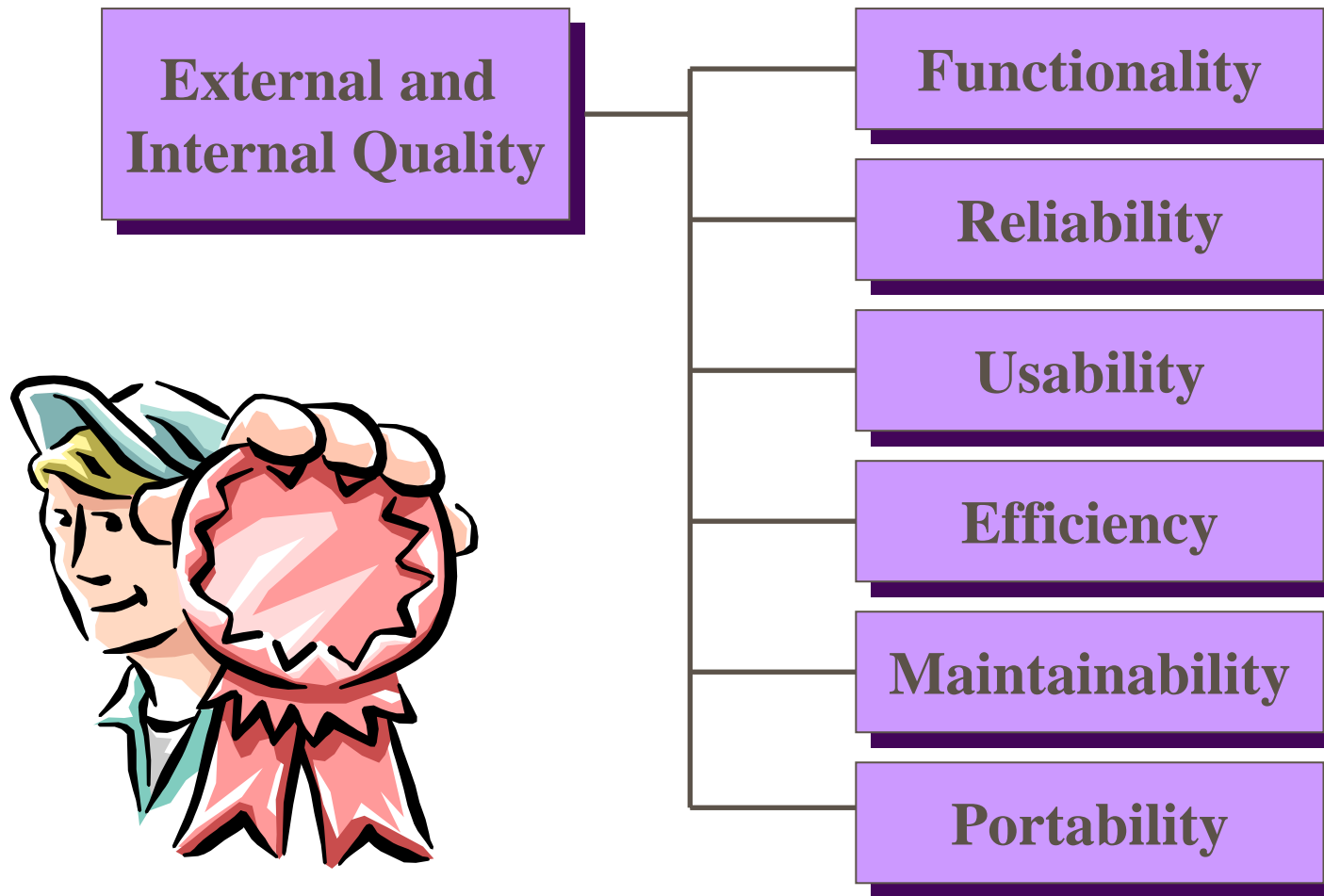


INTERNAL QUALITY DEFINITION

- **The totality of attributes of a product that determine its ability to satisfy stated and implied needs when used under specified condition. (ISO/IEC 14598-1)**

ISO/IEC 9126-1 Quality Model (1/2)

External and Internal Quality



ISO/IEC 9126-1 - Quality Model

Quality Characteristics

Subcharacteristics

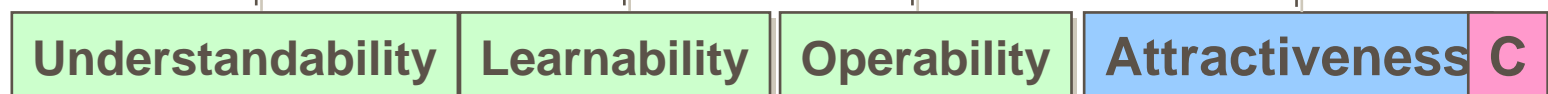
•Functionality



•Reliability



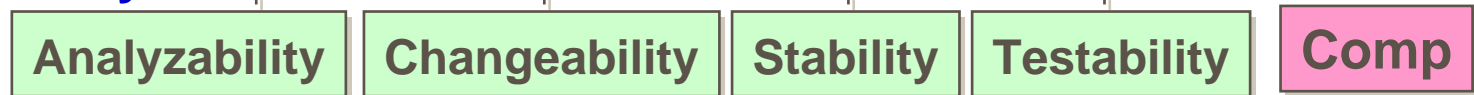
•Usability



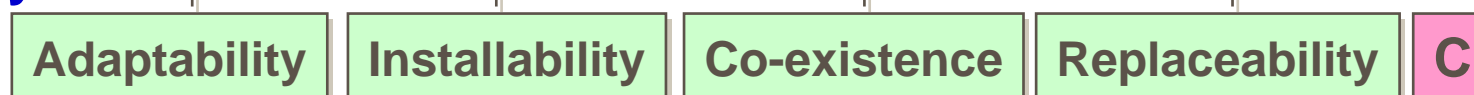
•Efficiency



•Maintainability



•Portability



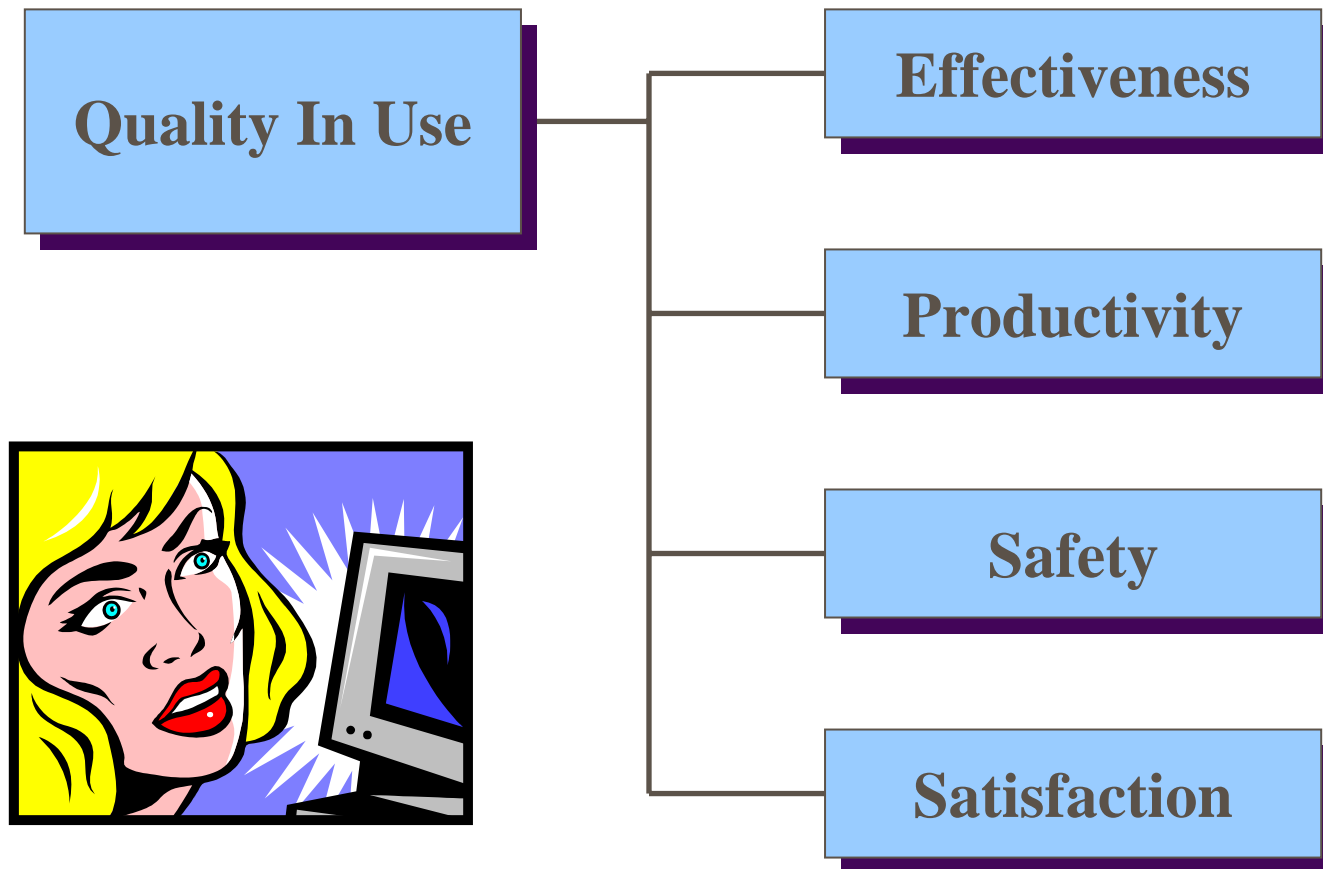


QUALITY IN USE DEFINITION

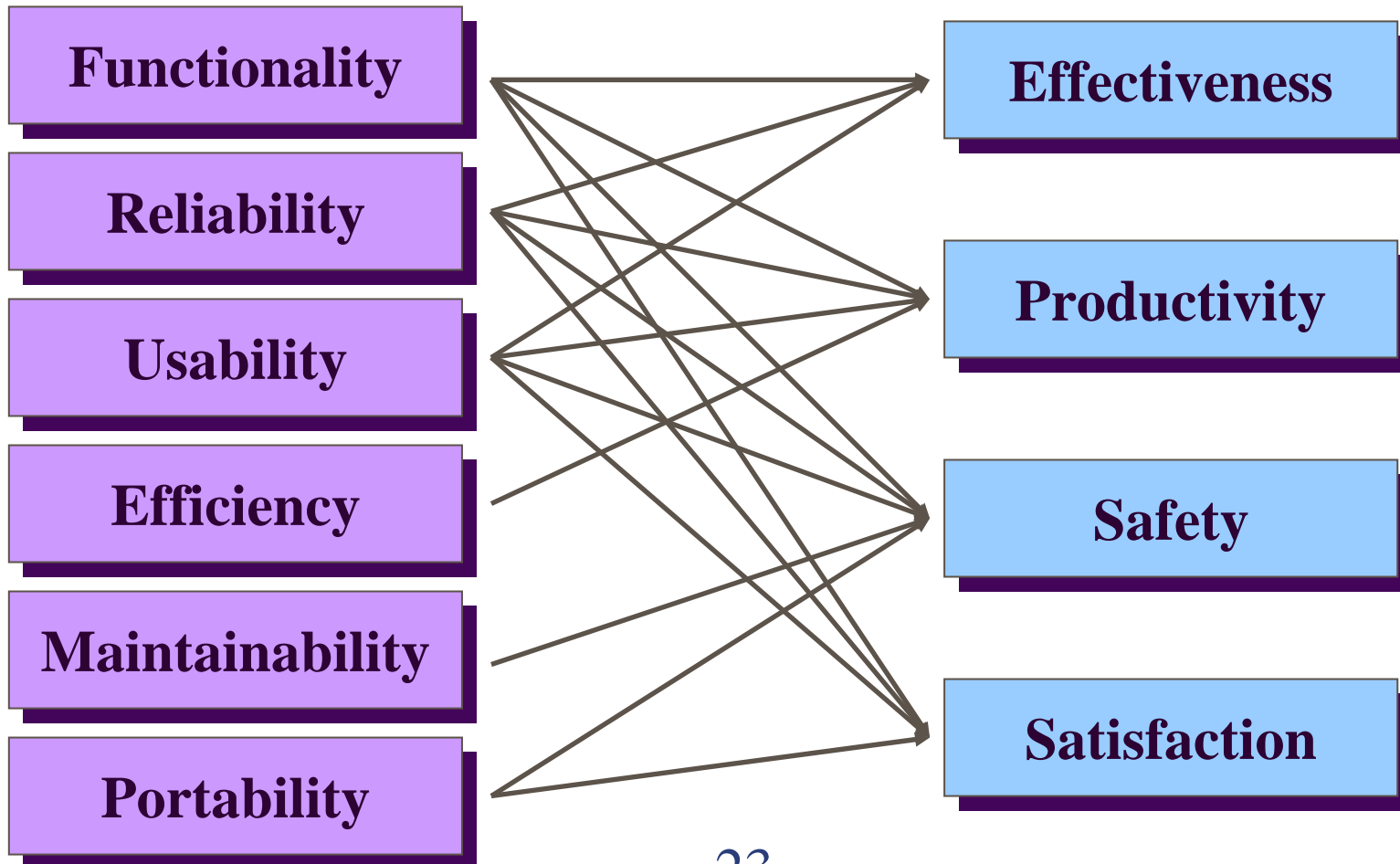
- **The extent to which a product used by specified users meets their needs to achieve specified goals with effectiveness, productivity and satisfaction in specified context of use. (ISO/IEC 14598-1)**

ISO/IEC 9126-1 Quality Model (2/2)

Quality In Use



Relationship Between Ext. / Int. Quality and Quality In Use





Part 3:
QUALITY LIFE CYCLE and
METHODOLOGIES

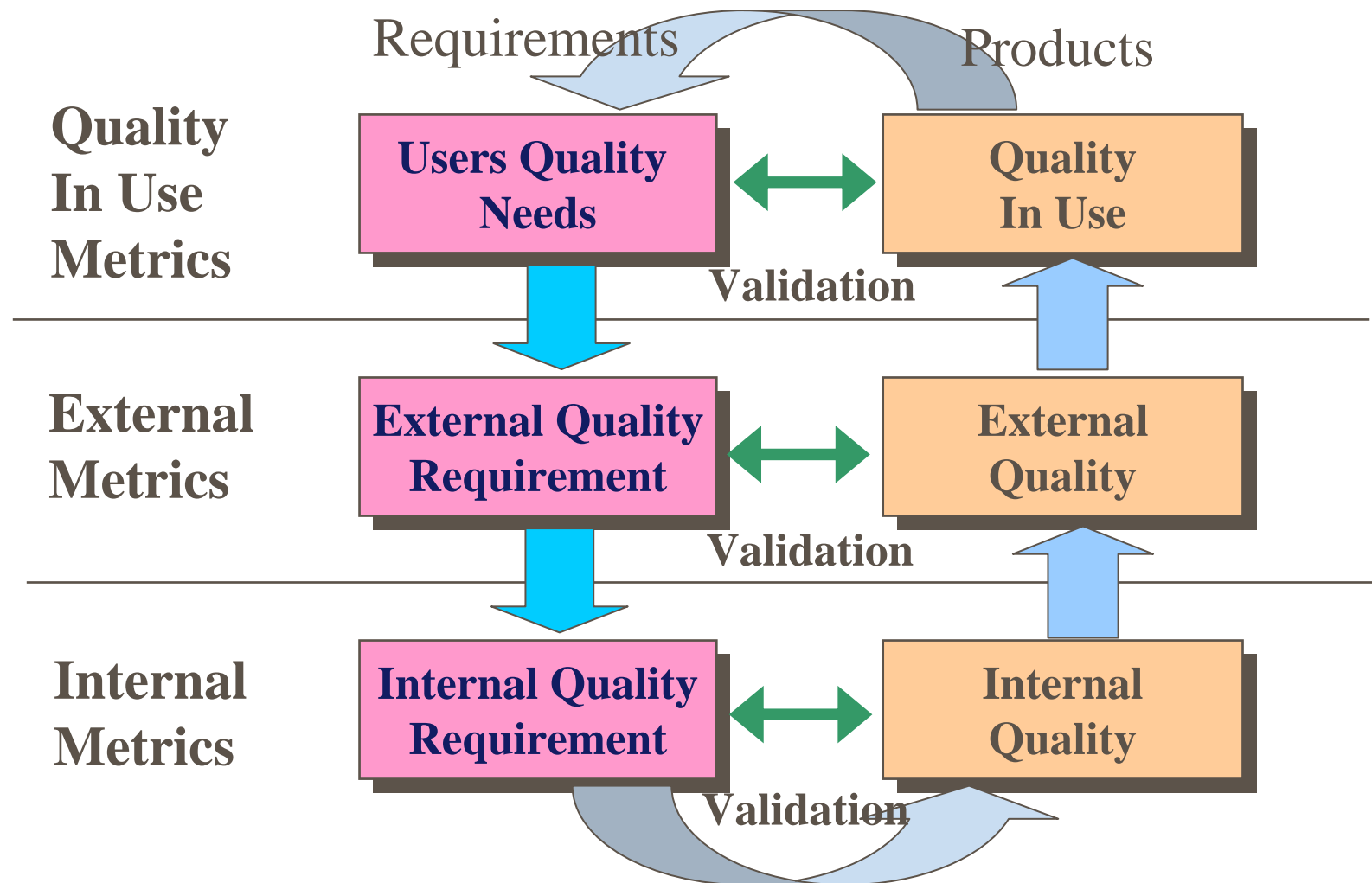
ISO/IEC 14598-1, 3, 4, 5

Requirements and Quality in Use

- “Needs” is expectations for the effects of a product.
- Requirements are identified or specified needs.
- A product to meet requirements is not always a good product when actually used (Quality In Use).



Quality in the Software Life-cycle





QUALITY IN USE REQUIREMENTS

- Quality In Use Varies Depending on a Context of Use.
- Context of Use = {User, End User, Environment in which Software is Used, Task, Operation}
- Results of Use = {Quality of the Output, Effect on Users, Effect on End Users, Effect on Environment of the System}
- Quality in Use Requirement can be identified by scenario of use.



EXTERNAL QUALITY REQUIREMENTS

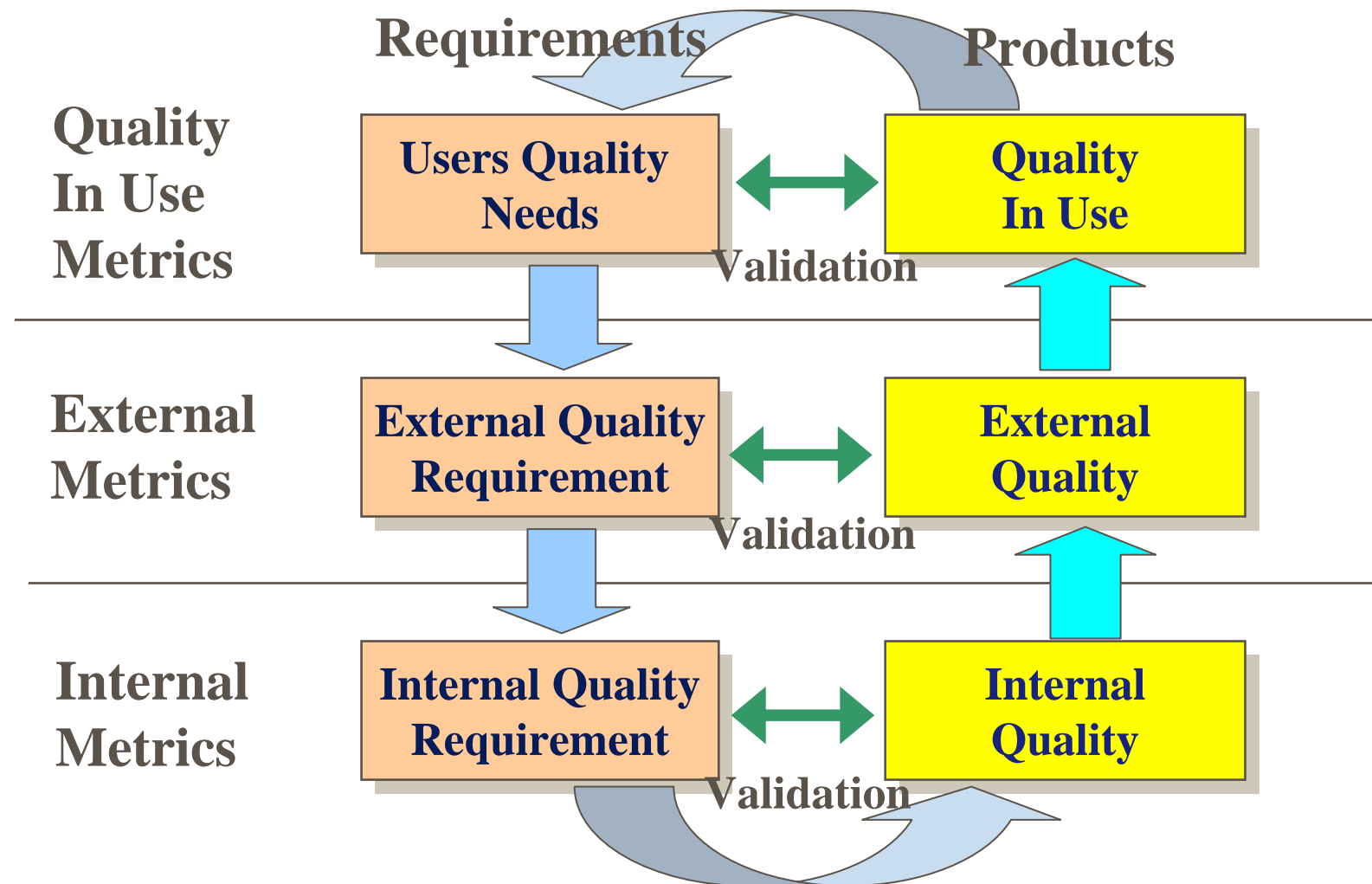
- **Quality Requirement** shall be stated at planning and requirements stage.
- A Standard **Quality Model** is helpful as a baseline for Quality Requirements Check List.
- Quality Requirement should be stated precisely and quantitatively using **quality measures**.
- Requirement Definition Technique should support to state Functional Requirements as well as all Quality Requirements.



INTERNAL QUALITY REQUIREMENTS

- Internal Quality Requirement is a kind of Project Milestone from the view of Quality.
- In order to achieve the required External Quality, it is necessary to transfer it into Internal Quality Requirement at the Design Stage.
- Internal Quality Requirements can be categorized into Design Quality and Code Quality.
- Design Quality is more important.

Quality in the Software Life-cycle





INTERNAL QUALITY EVALUATION

- Evaluation during Design Review and Code Review.
- Target entities are Specifications or Source Codes.
- Target software is not executable yet.
- Two objectives of Internal Quality Evaluation are;
 - **To clarify whether the internal quality requirements are satisfied**
 - **To predict the product quality when developed**



EXTERNAL QUALITY EVALUATION

- **EQE is Validation aiming to clarify that the All Quality Requirements are satisfied.**
- **Evaluation during Testing Stage.**
- **Evaluate the software using it as a System, under Simulated Environment, by Test Data and Operators for the testing.**
- **Results of the test must be recorded and summarized by using external metrics.**



QUALITY IN USE EVALUATION

- Evaluation after delivery aiming to use for revision up or other similar projects.
- Evaluation by actual users, in the real environment and real data.
- Some Needs which were NOT stated may be Identified.
- Method of Measurement are;
 - Feed-back from the users by a Questionnaire,
 - Observation of Users Behavior, or
 - Other measurement on site.



Part 4: **METRICS and MEASUREMENT**

ISO/IEC 9126-2, 3, 4

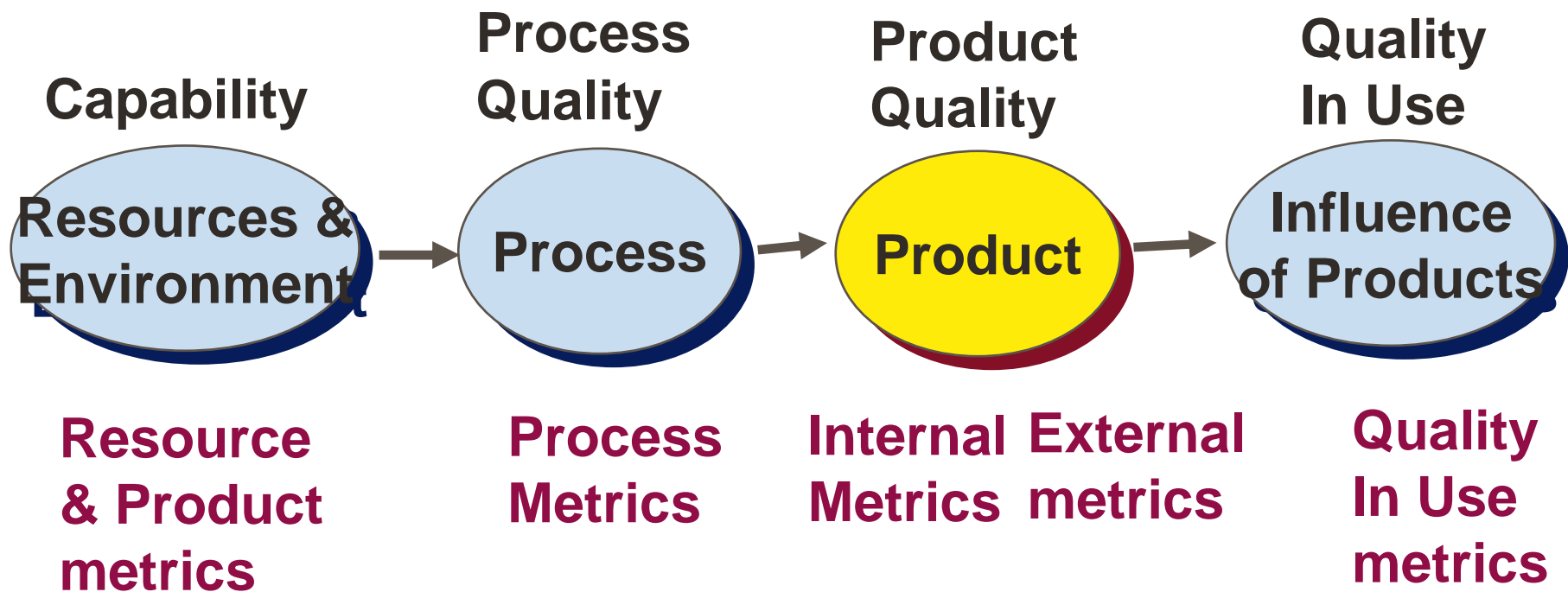


Why Measurement is Important?

- **Any product can be evaluated scientifically only when it's attributes are measured.**
- **In order to Improve Product Quality, its Attributes Must be Measured.**
- **Metrics should be Objective, Empirical and Reproductive.**

MEASUREMENT TARGETS AND METRICS

- A Target to be Measured and a Target to be evaluated may be different.





PRODUCT METRICS

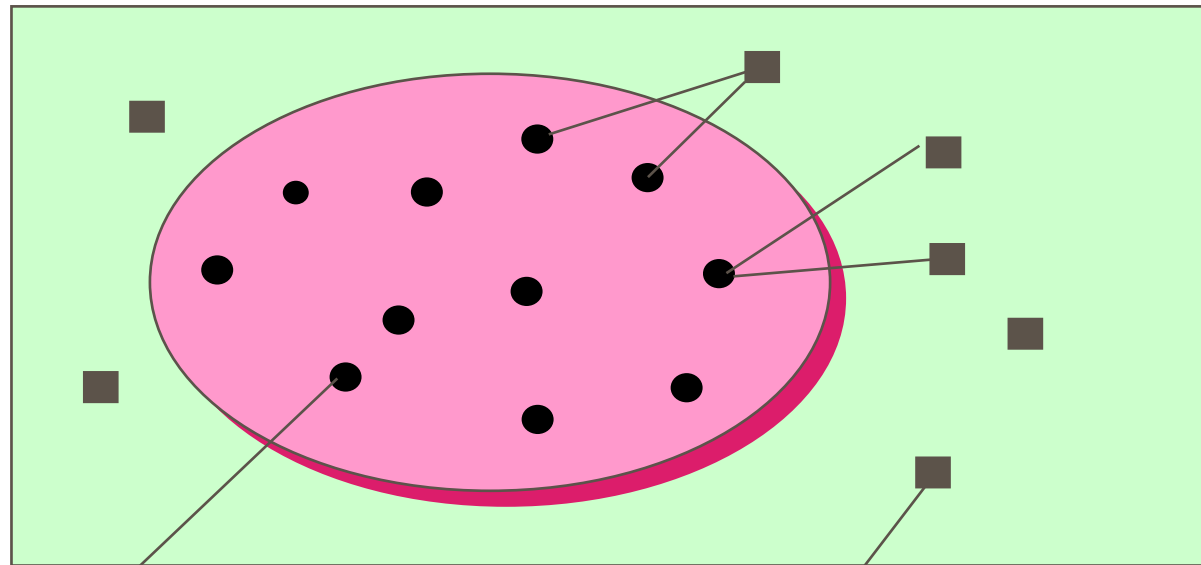
■ Purpose of Product Metrics

- To define quality requirements
- To measure and improve interim product quality
- To predict and control product quality
- To make decision on delivery or acceptance

■ Types of Product Metrics

- Internal Metrics
- External Metrics
- Quality in Use Metrics

Internal and External Attributes



**Internal Attributes =
Attributes of Software**

**External Attributes =
Attributes of System Behavior**



External Metrics Example: Functionality - Suitability (1)

- **Functional implementation**

completeness: $X = 1 - (A / B)$

A = Number of missing functions detected
in evaluation

B = Number of functions described in
requirement specifications



External Metrics Example: Functionality - Suitability (2)

- Functional specification stability (volatility)
- $X = 1 - A / B$
- A = Number of Functions Changed during and after System Testing Stage
- B = Number of Functions Described in Requirement Specification (or any function size measure) Example



External Metrics Example: Functionality - Security

- **Access Controllability**

- Illegal access detection ratio

$$X = A / B$$

- A = Number of Detected Illegal Operations

- B = Number of Illegal Operations
Anticipated in Specification



Internal Metrics Example:
Functionality - Suitability

■ **Functional Implementation**
Completeness

■ $X = 1 - A / B$

■ A = Number of **missing functions** detected in evaluation.

■ B = Number of **functions described in requirement specifications**

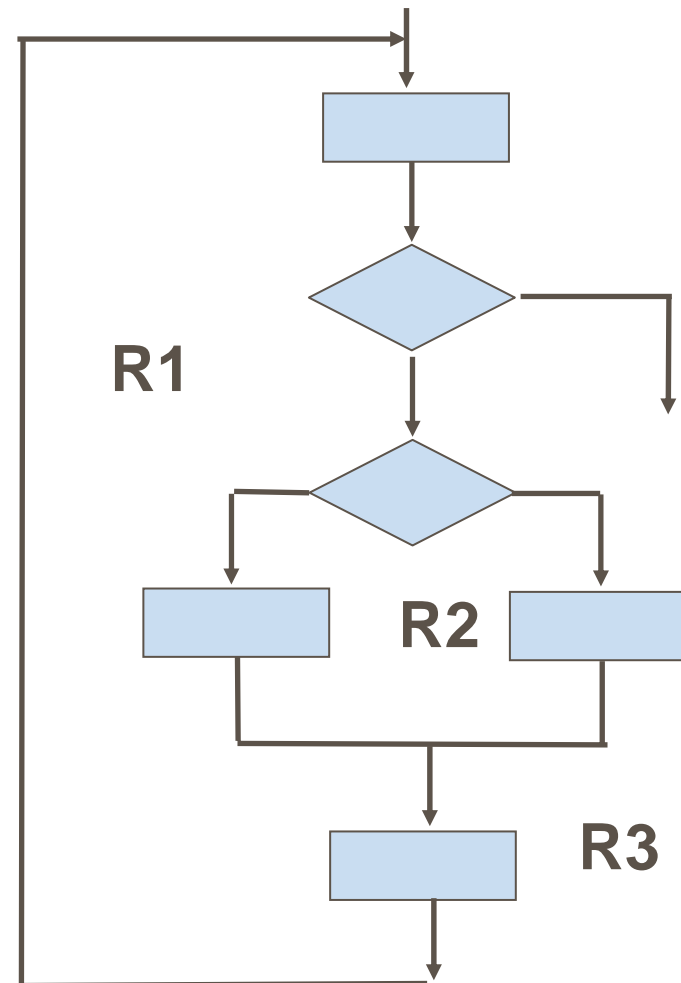


Internal Metrics Example: Usability - Operability

- **Messages Comprehensiveness**
- $X = A / B$
- A = Number of messages giving a **clear explanation and instructions** for action.
- B = **Number of messages** to be provided to the users.

Pure Internal Metrics Example:

- **Cyclomatic Complexity**
- **Modularity Metrics**
 - Cohesion
 - Coupling
- **Structure Metrics**
 - Fan-In, Fan-Out
- **Source Code Metrics**
 - Indentation
 - Procedure Name





Quality In Use Metrics Example: Effectiveness

- Task Effectiveness
- Proportion of correctly achieved goals of the task.
- $M1 = |1 - \sum A_i|$
- A_i : Proportional value of each missing or incorrect component in the task output



Quality In Use Metrics Example: Safety

- User health and safety
- $X = 1 - A / B$
- A : Number of users who reported RSI
 - RSI: Repetitive Strain Injury, Fatigue, Headache, etc
- B : Total number of users



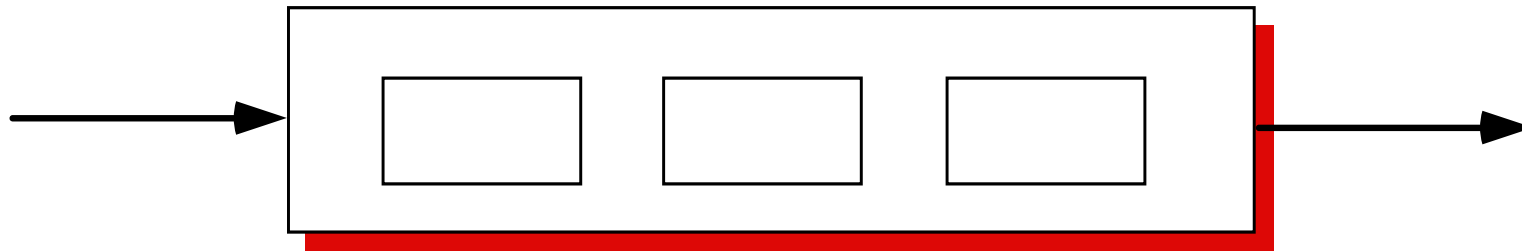
Part 5:
SOFTWARE PRODUCT
EVALUATION
PLANNING and MANAGEMENT

Purpose and Target Entity of Evaluation

Improve

Internal Environment

Environment Evaluation



Compare & Select

Method

Tool

Human Resources

Resource Evaluation

Control & Improve

Technical process

Managerial process

Process Evaluation

Decide Accept / Release

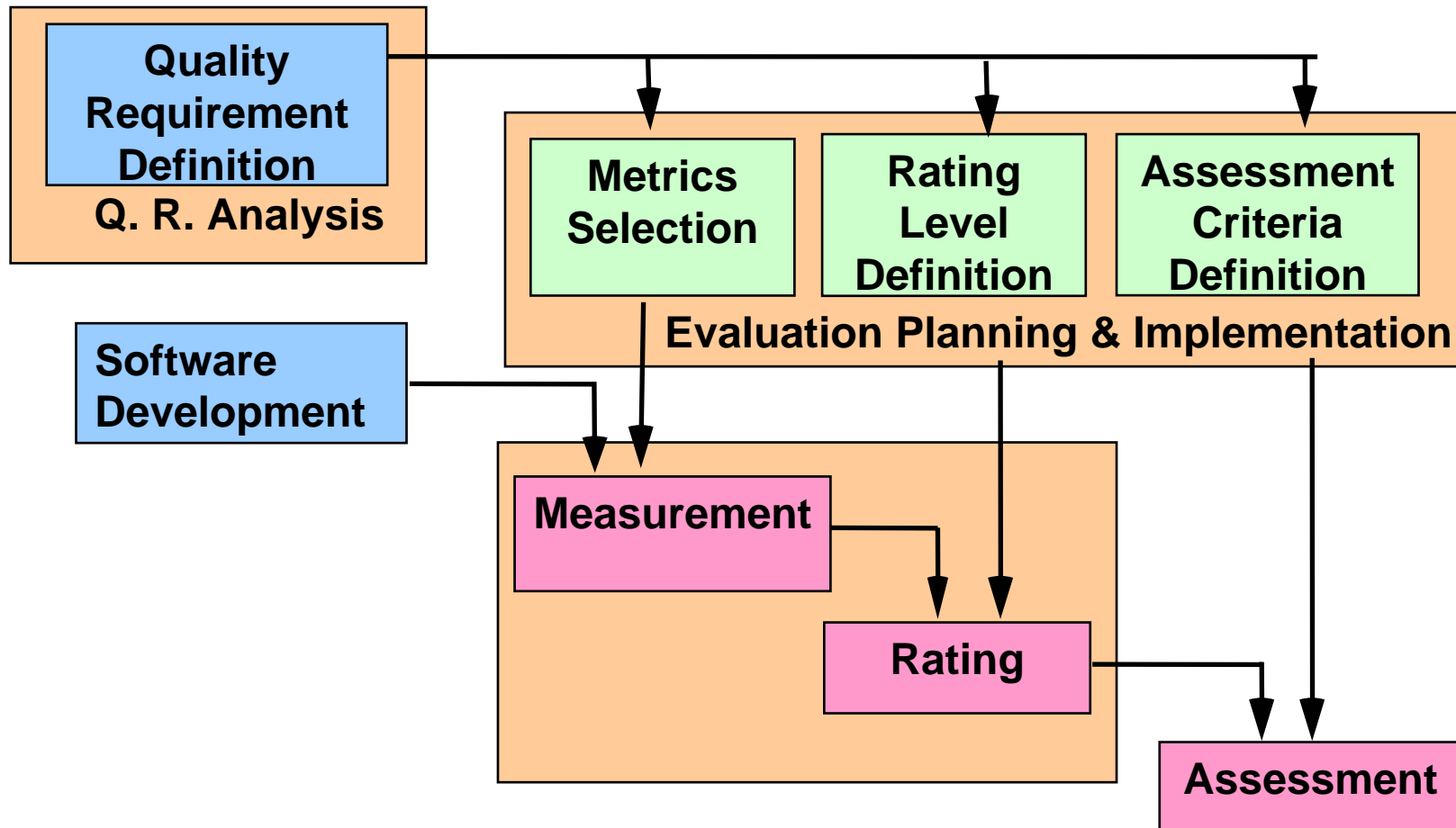
Input products

Intermediate products

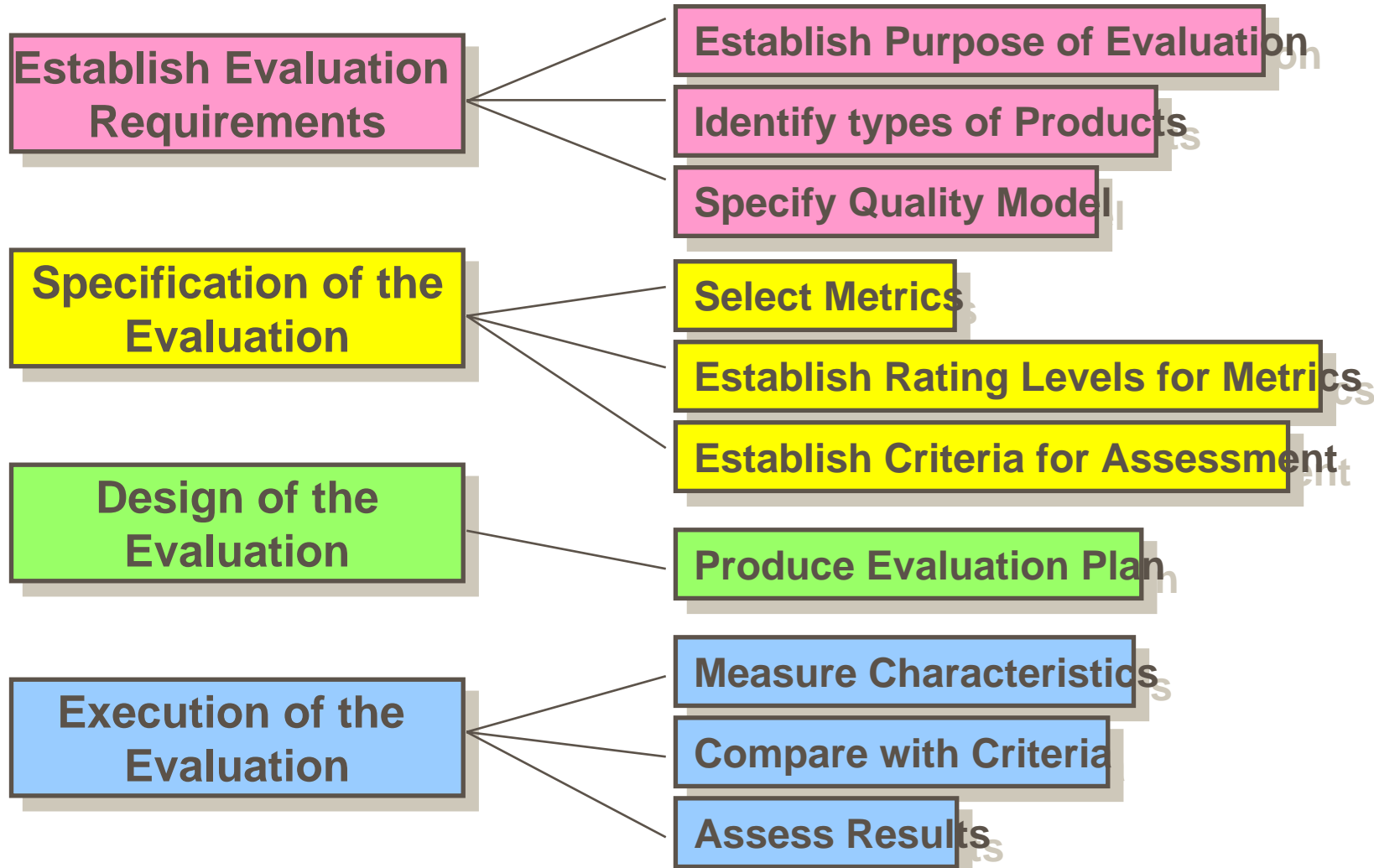
Output products

Product Evaluation

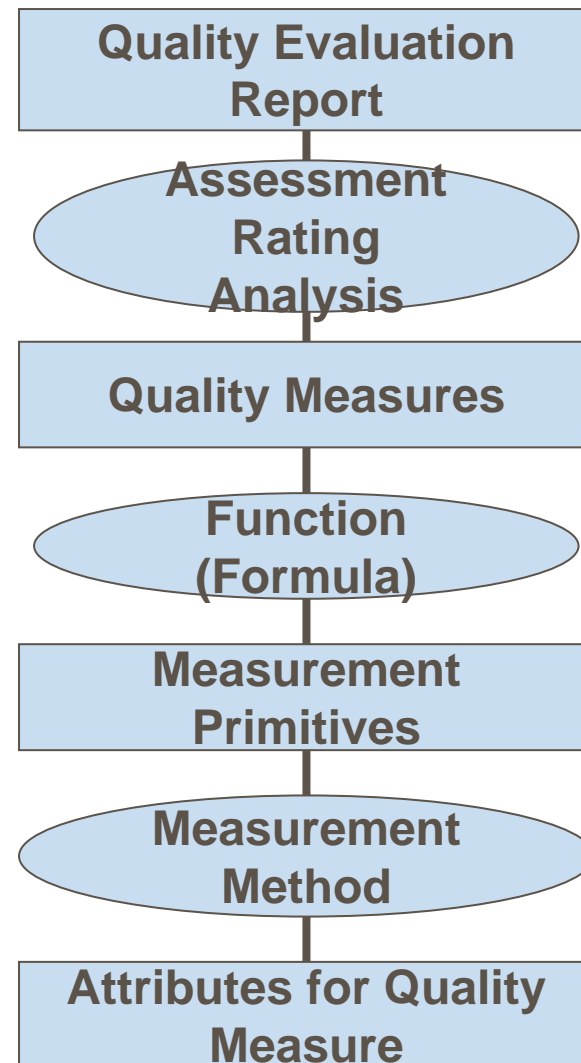
Evaluation Process (ISO/IEC 9126 - 1991)



Evaluation Process (ISO/IEC DIS14598-1)

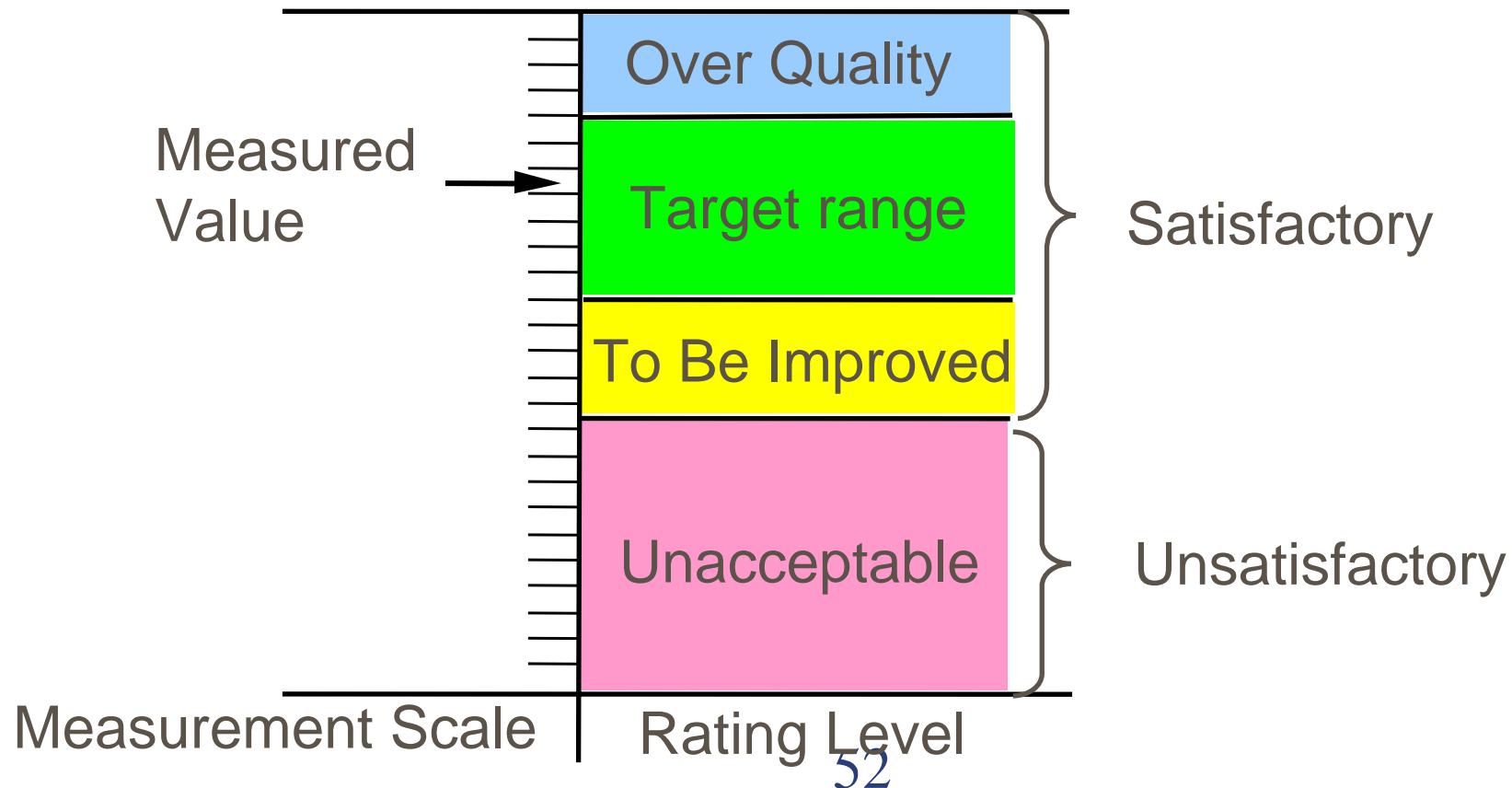


Quality Measurement Reference Model(ISO/IEC 25020)



Rating and Rating Levels

- To make judgement on how good the attribute is.
 - Normalize the measure (assigned value)





ASSESSMENT

- **Summarize measurement and rating results for every quality characteristic and subcharacteristic.**
- **Visualize the results.**
- **Review the assessment criteria and managerial requirement (Additional cost and delivery date).**
- **Consider the trade-off.**
- **Make such final decision as, to forward to the next process, to release the software, and to accept the software.**



QUALITY EVALUATION REQUIREMENTS






- **Repeatability**
- **Objectivity**
- **Quantitativeness**
- **Indicativeness**
- **Cost Effectiveness**
- **Inclusiveness**



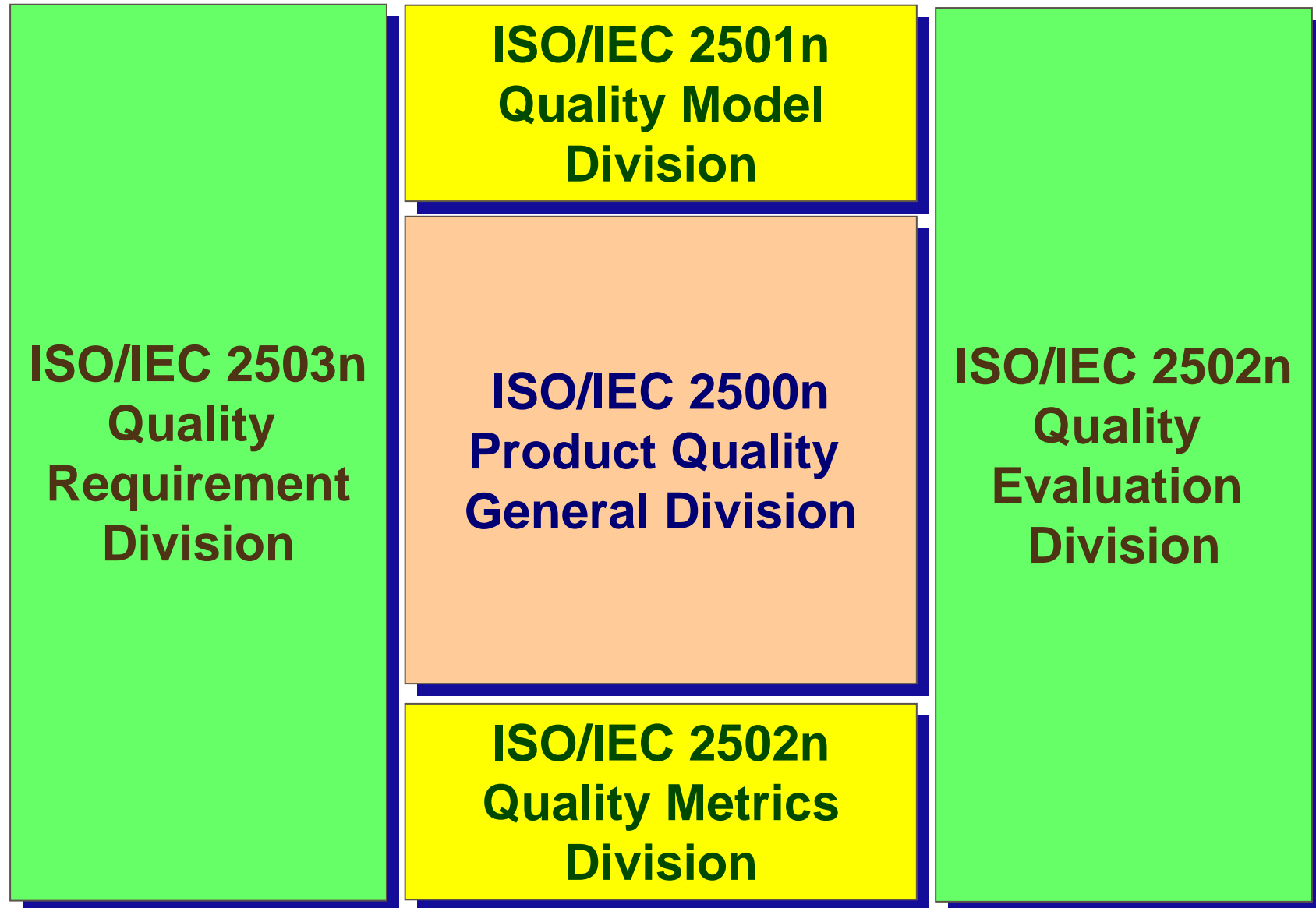
Produce Evaluation Planing

- **Major Items of a Evaluation Plan (ISO/IEC 14598-2)**
 - **Project Objectives**
 - **Applicable quality characteristics**
 - **List of Priority**
 - **Quality Objectives**
 - **Schedules**
 - **Definition of responsibilities**
 - **Measurement Categories**
 - **Using and Analyzing Data**
 - **Reporting**

Remaining Problems of the 9126 & 14598 series and SQuaRE

- Needs for **unique series name** and well organized numbers  ISO/IEC 25000 (SQuaRE Series)
- Needs for unique new **architecture** and **umbrella guide**  ISO/IEC 25000
- Needs for **guide for use of metrics**  ISO/IEC 25020
- Needs for Elementary **Metrics**  ISO/IEC 25021
- Needs for **Quality Requirement** standard  ISO/IEC 25030

SQuaRE: Architecture





ISSUES FOR THE FUTURE

■ **Technical Issues:**

- Integrate with CASE and Management Tools.
- Develop & Validate Measures.
- Invite More Field Experiences.

■ **Standard Issues:**

- Integrity within Software Engineering Standards.
- Speed Up the Standardization Process.
- Develop Measures Registration Scheme.



References

- **ISO/IEC 25000: Software engineering – Software product Quality Requirements and Evaluation (SQuaRE) – Guide to SQuaRE**
- **ISO/IEC 25020 – Measurement reference model and guide**
- **ISO/IEC 25021 – Measurement primitives**
- **ISO/IEC 25030 – Quality requirements**