SOFTWARE QUALITY ASSURANCE

SOFTWARE QUALITY ASSURANCE?

- A planned and systematic pattern of all actions necessary to provide adequate confidence that an item or product conforms to established technical requirements.[IEEE]

- A set of activities designed to evaluate the process by which the products are developed or manufactured. [IEEE]

SOFTWARE QUALITY ASSURANCE

- **Software quality assurance (SQA)** consists of means of monitoring the software engineering processes and methods used to ensure quality.

- SQA includes the entire software development process, which includes processes such as requirements definition, software design, coding, source code control, code reviews, change management, configuration management, testing, release management, and product integration.

WHAT IS SOFTWARE?

- IEEE:

Computer programs, procedures, and possibly associated documentation and data belongs to the operation of a computer system.

Software quality assurance always includes, in addition to code quality, the quality of the procedures, the documentation and the necessary software data.

ERROR, FAULT AND FAILURE

- **Error:** Are human interaction/dealings which produce an incorrect result.
- **Fault:** Fault is a stage of software which is caused by an error.
- **Failure:** It is a deviation of software from its expected delivery or service.
- **IEEE:**
  - **Error:** The difference between a computed, observed, or measured value or condition and the true, specified, or theoretically correct value or condition.
  - **Fault**: An incorrect step, process, or data definition.
  - **Failure**: An incorrect result.

9 CAUSES OF SOFTWARE ERRORS

- Faulty requirements definition
- Client-developer communication failures
- Considerable deviations from software requirements
- Logical design errors
- Coding errors
- Non-compliance with documentation and coding instructions
- Shortcomings of the testing process
- User interface and procedure errors
- Documentation errors

QUALITY?

- The degree to which a system, component, or process meets specified requirements. **[IEEE]**
- The degree to which a system, component, or process meets customer or user needs or expectations. **[IEEE]**
- The completeness of features and characteristics of a product or service that bear on its ability to satisfy specified or implied needs**.[ISO]**
- Conformance to explicitly stated functional and performance requirements, explicitly documented development standards, and implicit characteristics that are expected of all professionally developed software**.[Roger S. Pressman]**

SOFTWARE QUALITY FACTORS

- The software quality factors are those reasons which do affect quality of the software.

THE NEED FOR A COMPREHENSIVE DEFINITION OF REQUIREMENTS

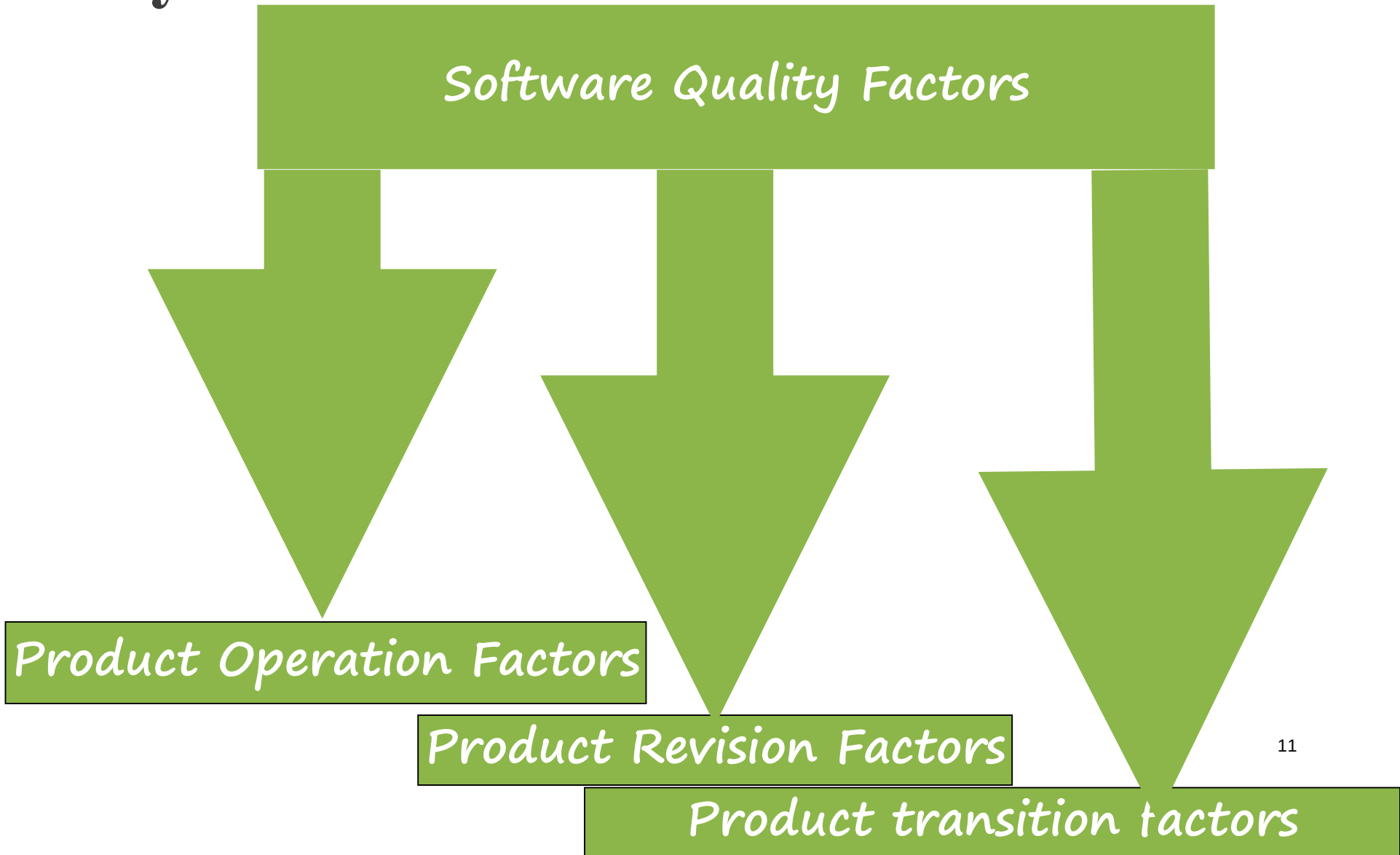
There is a need for a comprehensive definition of **REQUIREMENTS** that will cover all attributes of software and aspects in order to assure the full **SATISFACTION** of the users. The great variety of issues related to the various attributes of software and its use and maintenance, as **DEFINED IN SOFTWARE REQUIREMENTS DOCUMENTS**, can be

classified into **CONTENT GROUPS** called "***quality factors***"

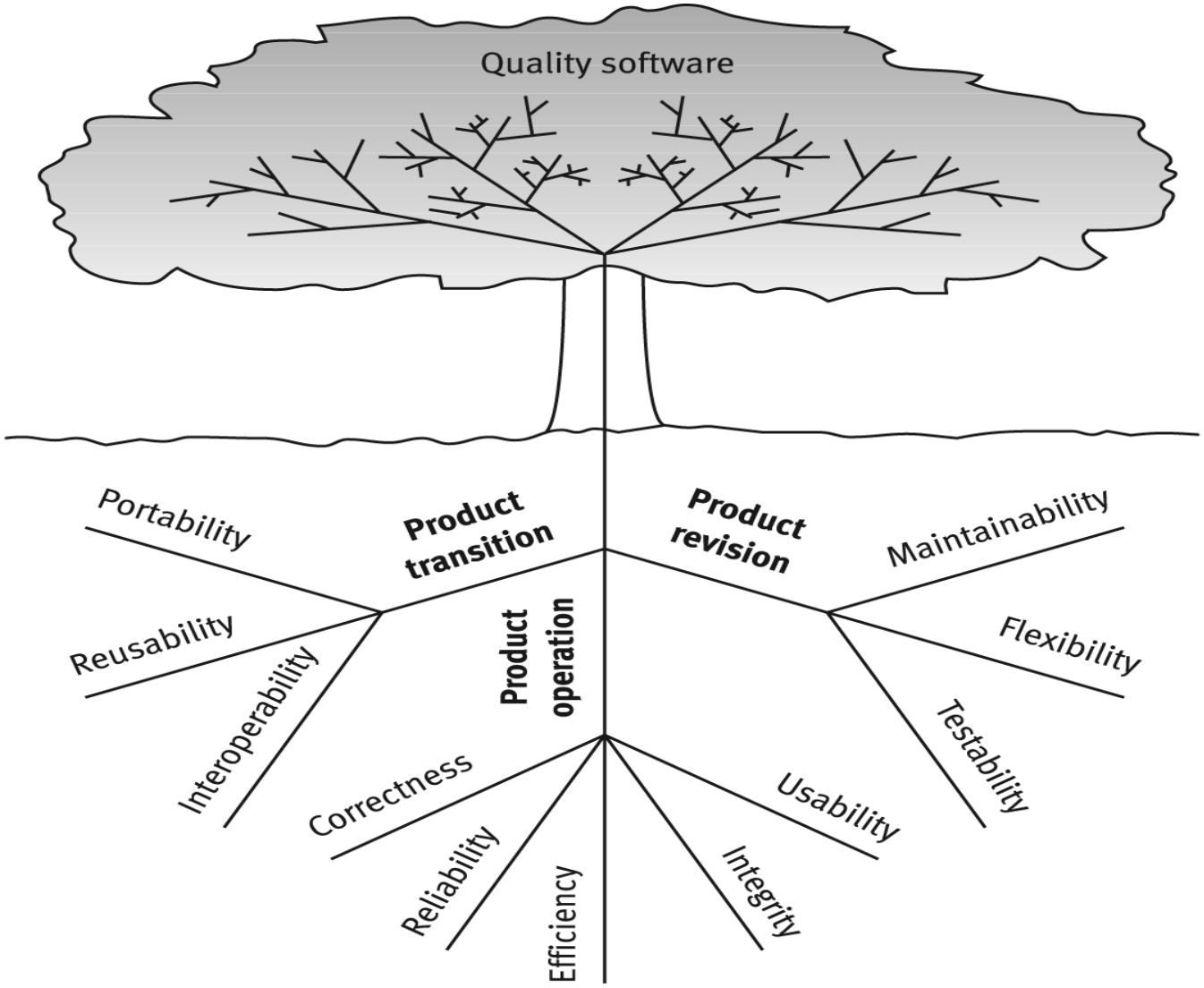
CLASSIFICATION OF SW REQUIREMENTS INTO QUALITY FACTORS

- Several models of software quality factors and their categorization in factor categories have been suggested over the years.
- **The classic model of software quality factors, suggested by McCall, consists of 11 factors.**

McCall's Quality Factors Model



McCall's Factor Model Tree



MCCALL'S QUALITY FACTORS MODEL

1. Product Operation Software Quality Factors:

According to model five software qualities are included in the product operation category which deal with requirements that directly affect the daily operation of the software.

SOFTWARE QUALITY FACTORS:

1. PRODUCT OPERATION SOFTWARE QUALITY FACTORS

- **Correctness:** Correctness requirements are defined in a list of the software system's required outputs, such as a query display of a customer's balance in the sales accounting information system(**Related to output**)
- **Reliability:** Reliability requirements deal with failures to provide service. They determine the maximum allowed software system failure rate, and can refer to the entire system or to one or more of its separate functions.(**Related to failures**)
- **Efficiency:** Efficiency requirements deal with the hardware resources needed to perform all the functions of the software system in conformance to all other requirements.(**Related to resources**)
- **Integrity:** Integrity requirements deal with the software system security, that is, requirements to prevent access to unauthorized persons, to distinguish between the majority of personnel allowed to see the information ("read permit") and a limited group who will be allowed to add and change data ("write permit"), and so forth.
- **Usability:** Usability requirements deal with the scope of staff resources needed to operate the software system and to train a new employee.

2.PRODUCT REVISION SOFTWARE QUALITY FACTORS

These factors deal with those requirements that affect the complete range of software maintenance activities:

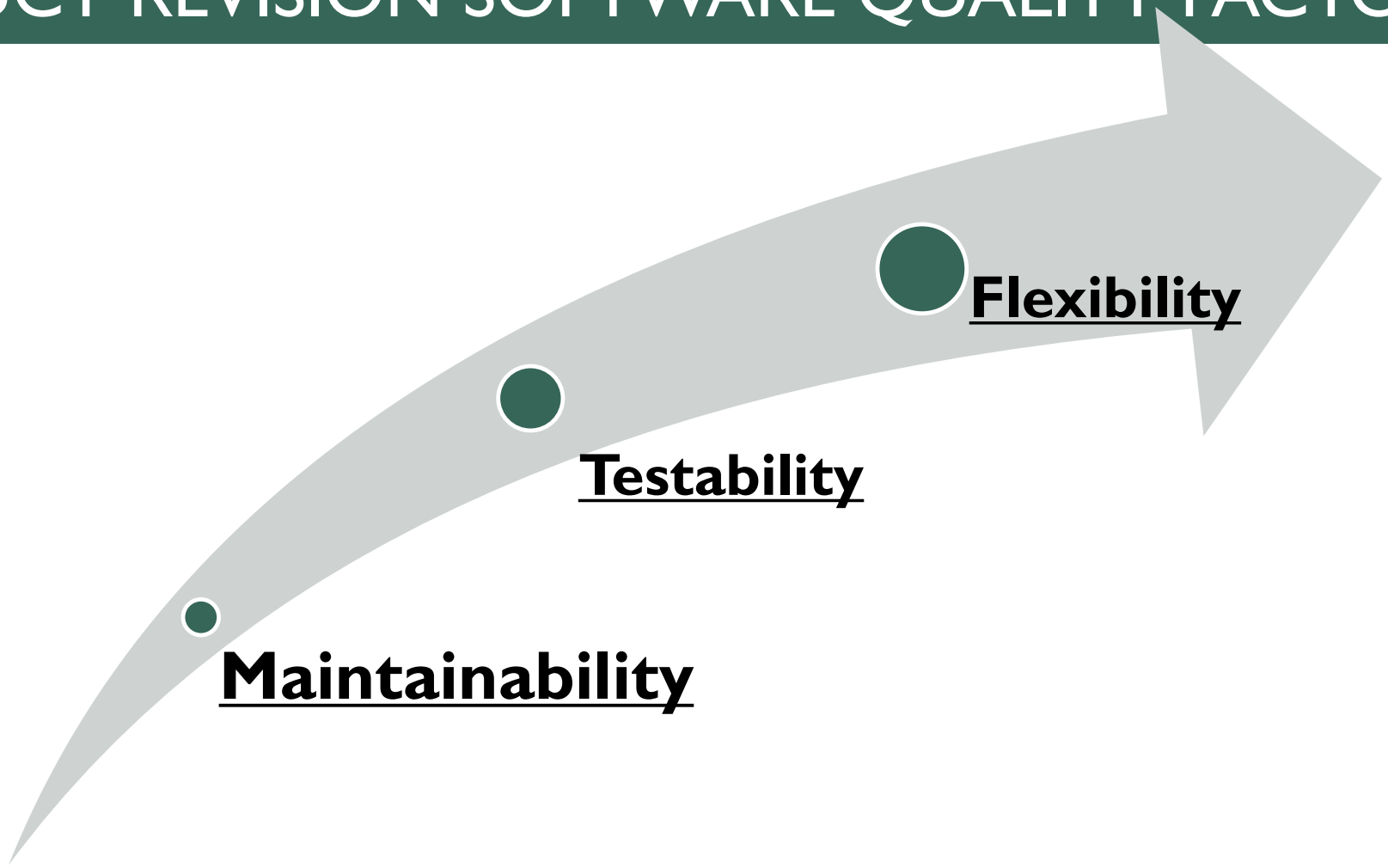
1. **Corrective maintenance** (correction of software faults and failures)
2. **Adaptive maintenance** (adapting the current software to additional circumstances and customers without changing the software)
3. **Perfective maintenance** (enhancement and improvement of existing software)

Testability

Flexibility

Maintainability

2.PRODUCT REVISION SOFTWARE QUALITY FACTORS



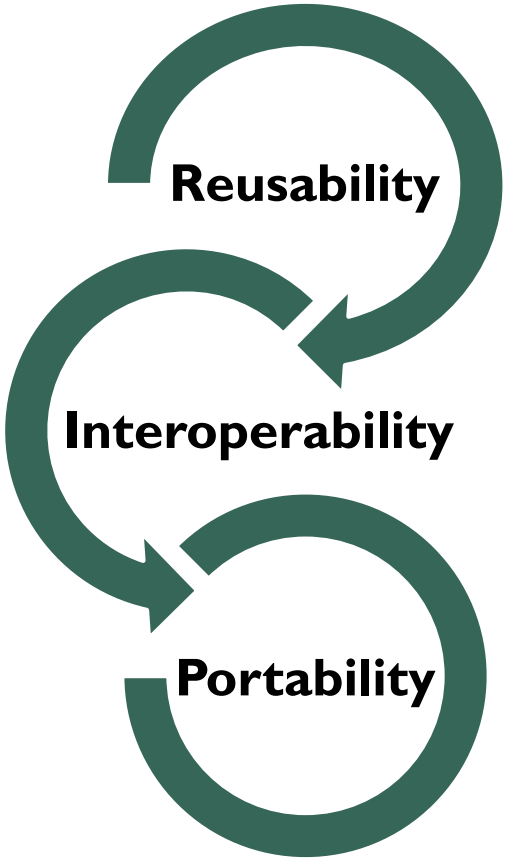
2. PRODUCT REVISION SOFTWARE QUALITY FACTORS

- **Maintainability :** Maintainability requirements determine the efforts that will be needed by users and maintenance people to identify the reasons for software failures, to correct the failures, and to verify the success of the corrections.
- **Flexibility :** The capabilities and efforts required to support adaptive maintenance activities are covered by the flexibility requirements. This factor's requirements also support perfective maintenance activities
- **Testability :** Testability requirements deal with the testing of an information system and its operation. Another type of these requirements deals with automatic diagnostic checks applied by the maintenance technicians to detect the causes of software failures.

3.PRODUCT TRANSITION SOFTWARE QUALITY FACTORS

Three quality factors are included in the product transition category, a category that apply to the adaptation of software to other environments and its interaction with other software systems.

3. PRODUCT TRANSITION SOFTWARE QUALITY FACTORS



3. PRODUCT TRANSITION SOFTWARE QUALITY FACTORS

- **Portability :** Portability requirements tend to the adaptation of a software system to other environments consisting of different hardware, different operating systems and so forth
- **Reusability:** Requirements deal with the use of software modules originally designed for one project in a new software project currently being developed.
- **Interoperability :** Interoperability requirements focus on creating interfaces with other software systems or with other equipment firmware (for example, the firmware of the production machinery and testing equipment interfaces with the production control software).