Course 5

Software Quality Models

Software Quality Models

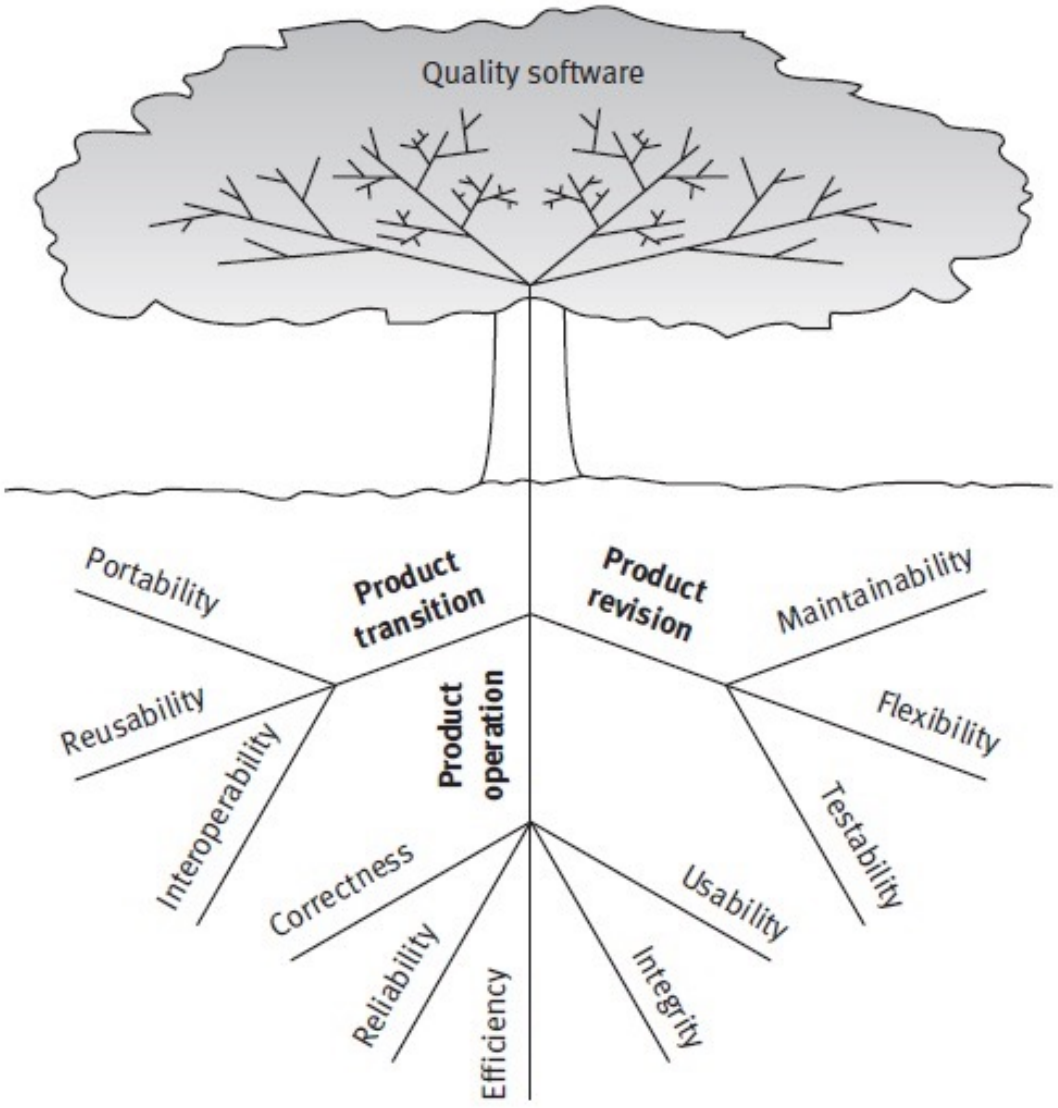
- McCall (1977) – classic – 11 factors
- Evans & Marciniak (1987) – 12 factors
- Deutsch & Willis (1988) – 15 factors
- ISO 9126 (2001) – 6 factors
- ISO25010 (2011) – 9 factors
- ISO25010 (2023) – 10 factors

Terminology

- Hierarchical structure: set of factors with subfactors
- Factor/ subfactor = characteristics / subcharacteristics
- In McCall model – same subfactor part of different factors
- Rest of the models: tree structure

McCall Model

- 11 factors – grouped in 3 categories:
  - Product operation factors - deals with requirements that directly affects software operation: Correctness, Reliability, Efficiency, Integrity, Usability
  - Product revision factors - deals with requirements affecting software maintenance activities: Maintainability, Flexibility, Testability
  - Product transition factors - deals with requirements affecting adaptation and integration: Portability, Reusability, Interoperability


Quality software
Portability
Product transition
Product revision
Maintainability
Reusability
Interoperability
Product operation
Flexibility
Testability
Correctness
Reliability
Efficiency
Integrity
Usability

Other models

Evans & Marciniak

- Exclude testability
- Add verifiability
- Add expandability

Deutsch & Willis

- Exclude testability
- Add verifiability
- Add expandability
- Add safety
- Add manageability
- Add survivability

➢Expandability + survivability – resemble flexibility + reliability
➢Testability – part of maintainability

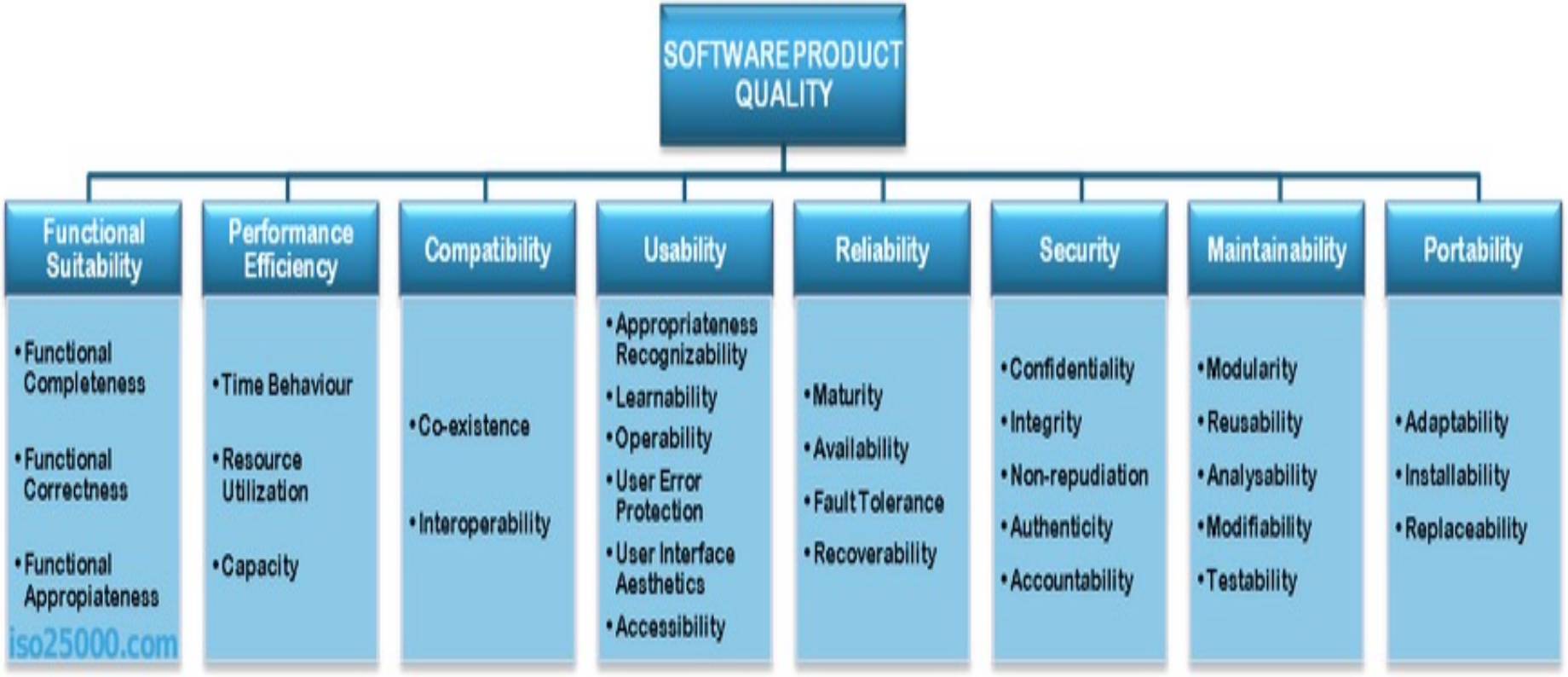
FURPS Model

- Developed by Hewlett Packard
- **FURPS:**
  - **Functionality -** is assessed by evaluating the features and capabilities of the delivered program and the overall security of the system.
  - **Usability -** is assessed by considering human factors, overall aesthetics, look and feel and easy of learning.
  - **Reliability -** is assessed by measuring the frequency of failure, accuracy of output, the mean-time-to-failure(MTTF), ability to recover from failure.
  - **Performance -** is assessed by processing speed, response time, resource utilization, throughput and efficiency.
  - **Supportability -** is assessed by the ability to extend the program (extensibility), adaptability, serviceability and maintainability.

ISO 9126 Quality Factors

- The ISO 9126 standard identifies six key quality attributes:
  - **Functionality -** degree to which software satisfies stated needs.
  - **Reliability -** the amount of time the software is up and running.
  - **Usability -** the degree to which a software is easy to use.
  - **Efficiency -** the degree to which software makes an optimum utilization of the resources.
  - **Maintainability -** the ease with which the software can be modified.
  - **Portability -** the ease with which a software can be migrated from one environment to the other.

ISO 25010
System and software quality model



Evolution of SQ Models

ISO 9126

Functionality	Reliability	Usability	Efficiency	Maintainab.	Portability
Suitability	Maturity	Understand	Time behav.	Analysability	Adaptability
Accurateness	Fault tolerance	Learnability	Resource beh.	Changebility	Instability
Interoperability	Recoverability	Operability		Stability	Conformance
Compliance				Testability	Replacebility
Security					

ISO 25010

Func. Suit.	Perform. Efficency	Compatib.	Usability	Reliability	Security	Maintai-nability	Portability
Func. Compl.	Time behav	Co-existen.	Appropiat.	Maturity	Confid.	Modularity	Adaptab.
Func. Corect.	Resource util.	Interopera bility	Learna-bility	Availability	Integrity	Reusability	Instability
Func. Aprro.	Capacity		Operability	Fault tolerance	Non-repudianc e	Analysa-bility	Replace-bility
			User error prot.	Recovera-bility	Authenti-city	Modifia-bility	
			UI Asethet.		Accountab.	Testability	
			Accesibil.				

Evolution of SQ Models

ISO 9126

Functionality	Reliability	Usability	Efficiency	Maintainab.	Portability
Suitability	Maturity	Understand	Time behav.	Analysability	Adaptability
Accurateness	Fault tolerance	Learnability	Resource beh.	Changebility	Instability
Interoperability	Recoverability	Operability		Stability	Conformance
Compliance				Testability	Replacebility
Security					

ISO 25010

Func. Suit.	Perform. Efficency	Compatib.	Usability	Reliability	Security	Maintai-nability	Portability
Func. Compl.	Time behav	Co-existen.	Appropiat	Maturity	Confid.	Modularity	Adaptab.
Func. Corect.	Resource util.	Interopera bility	Learna-bility	Availability	Integrity	Reusability	Instability
Func. Aprro.	Capacity		Operability	Fault tolerance	Non-repudianc e	Analysa-bility	Replace-bility
			User error prot.	Recovera-bility	Authenti-city	Modifia-bility	
			UI Asethet.		Accountab.	Testability	
			Accesibil.				

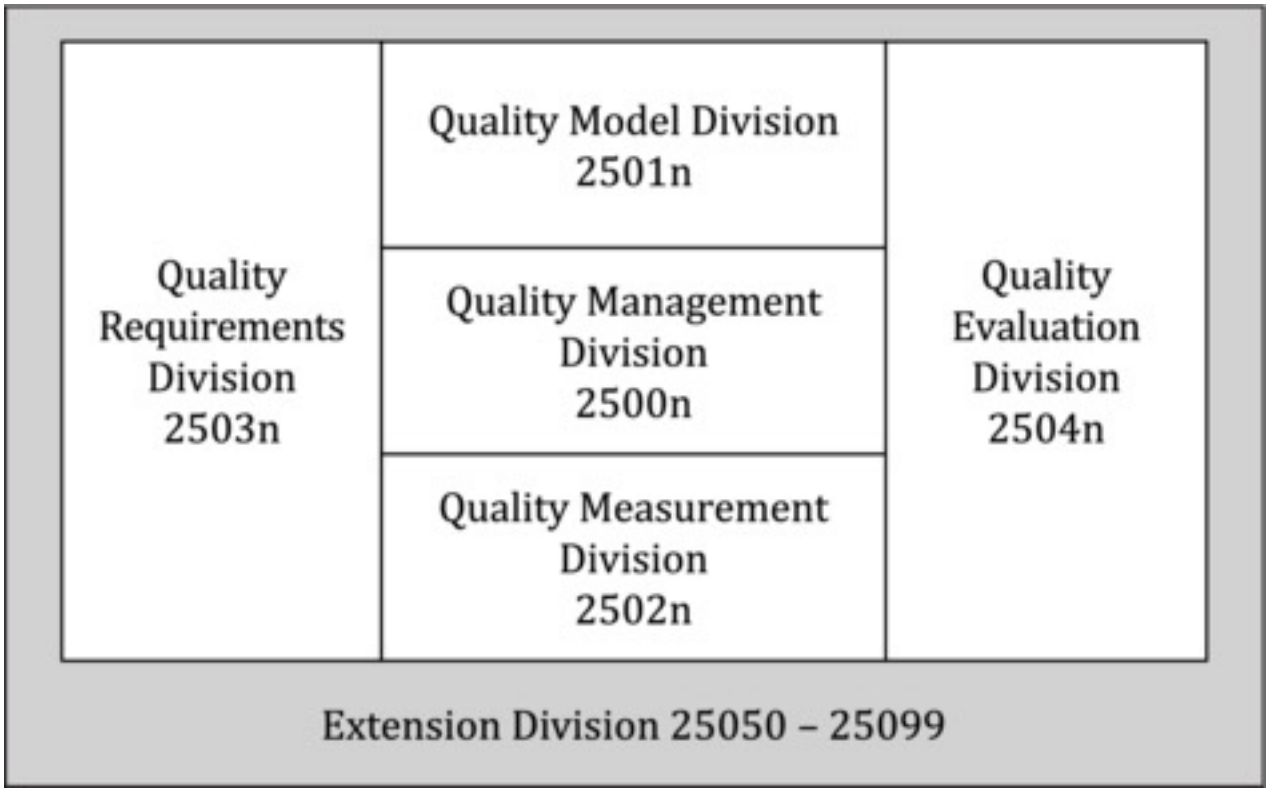
ISO 25010: 2011 vs 2023

ISO 25010: 2011

Func. Suit.	Perform. Efficency	Compatib.	Usability	Reliability	Security	Maintai-nability	Portability
Func. Compl.	Time behav	Co-existen.	Appropiat.	Maturity	Confid.	Modularity	Adaptab.
Func. Corect.	Resource util.	Interoperability	Learna-bility	Availability	Integrity	Reusability	Instability
Func. Aprro.	Capacity		Operability	Fault tolerance	Non-repudiance	Analysa-bility	Replace-bility
			User error prot.	Recovera-bility	Authenti-city	Modifia-bility	
			UI Asethet.		Accountab.	Testability	
			Accesibil.				

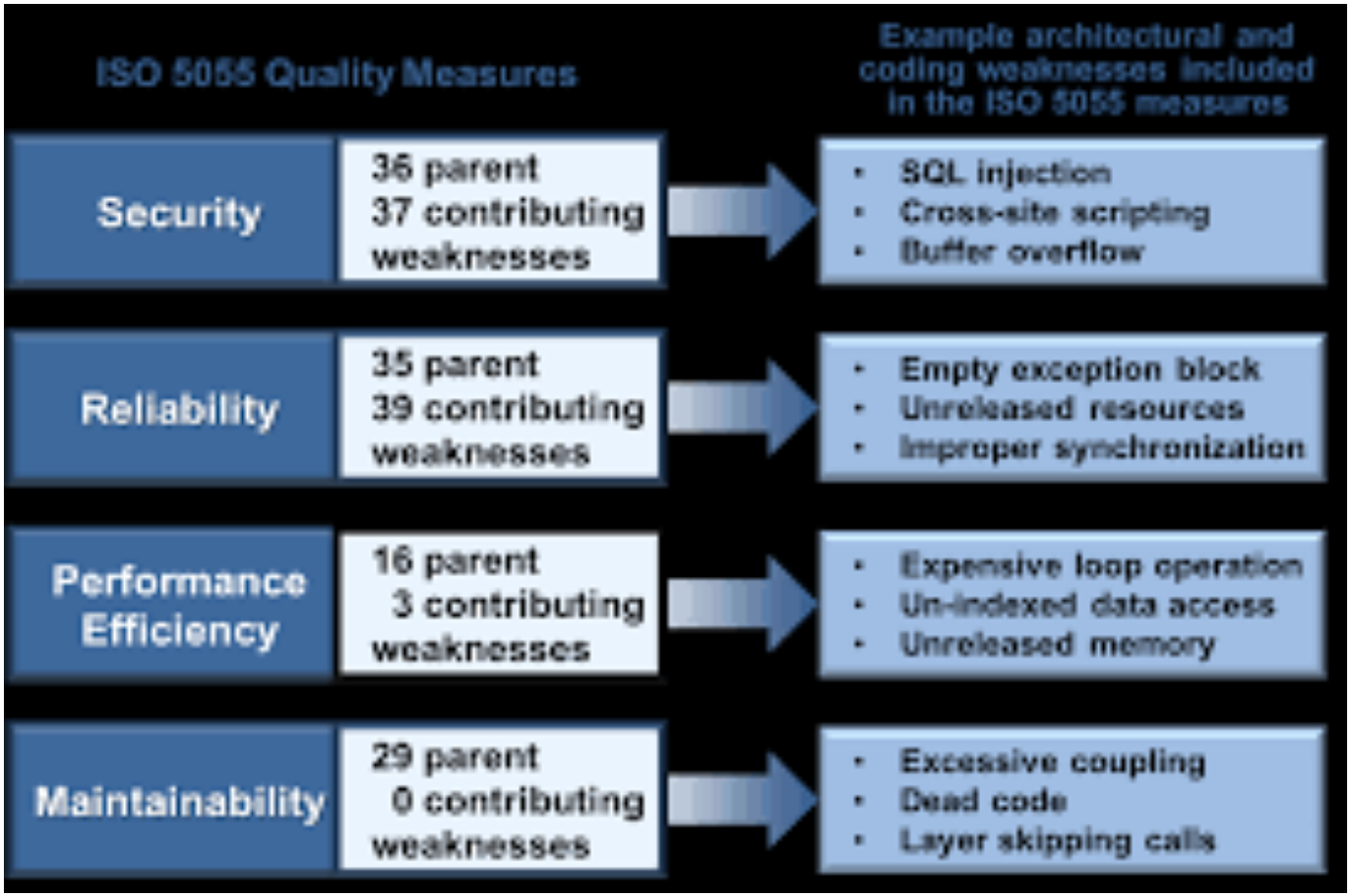
ISO 25010: 2023

Func. Suit.	Perform. Efficency	Compatib	interaction capability	Reliability	Security	Maintai-nability	Flexibility	Safety
Func. Compl.	Time behav	Coexist.	Appropiat.	faultlessness	Confid.	Modularity	Adaptab.	Oper. Constr.
Func. Corect.	Resour. util.	Interoper	Learnability	Availability	Integrity	Reusability	Scalability	Risk Identif.
Func. Aprro.	Capacity		Operability	Fault tolerance	Non-repudiance	Analysa-bility	Instability	Fail safe
			User error prot.	Recoverability	Authenticity	Modifia-bility	Replacebility	Hazard warning
			User engagement		Accountab.	Testability		Safe integration
			Inclusivity		Resistance			
			User assistance					
			Self descript.					


Quality Model Division
2501n
Quality Requirements Division
2503n
Quality Management Division
2500n
Quality Evaluation Division
2504n
Quality Measurement Division
2502n
Extension Division 25050 – 25099

ISO5055: 2021 https://cwe.mitre.org/

Software quality measurement — Automated source code quality measures



McCall model

- See `McCall.pdf` in course directory
- Approach:
  - Determine a set of quality factors
  - Develop a set of criteria for each factor
  - Define metrics for each criterion
  - Validate metrics
  - Translate results into guidelines
- Start with ≈ 55 features - group

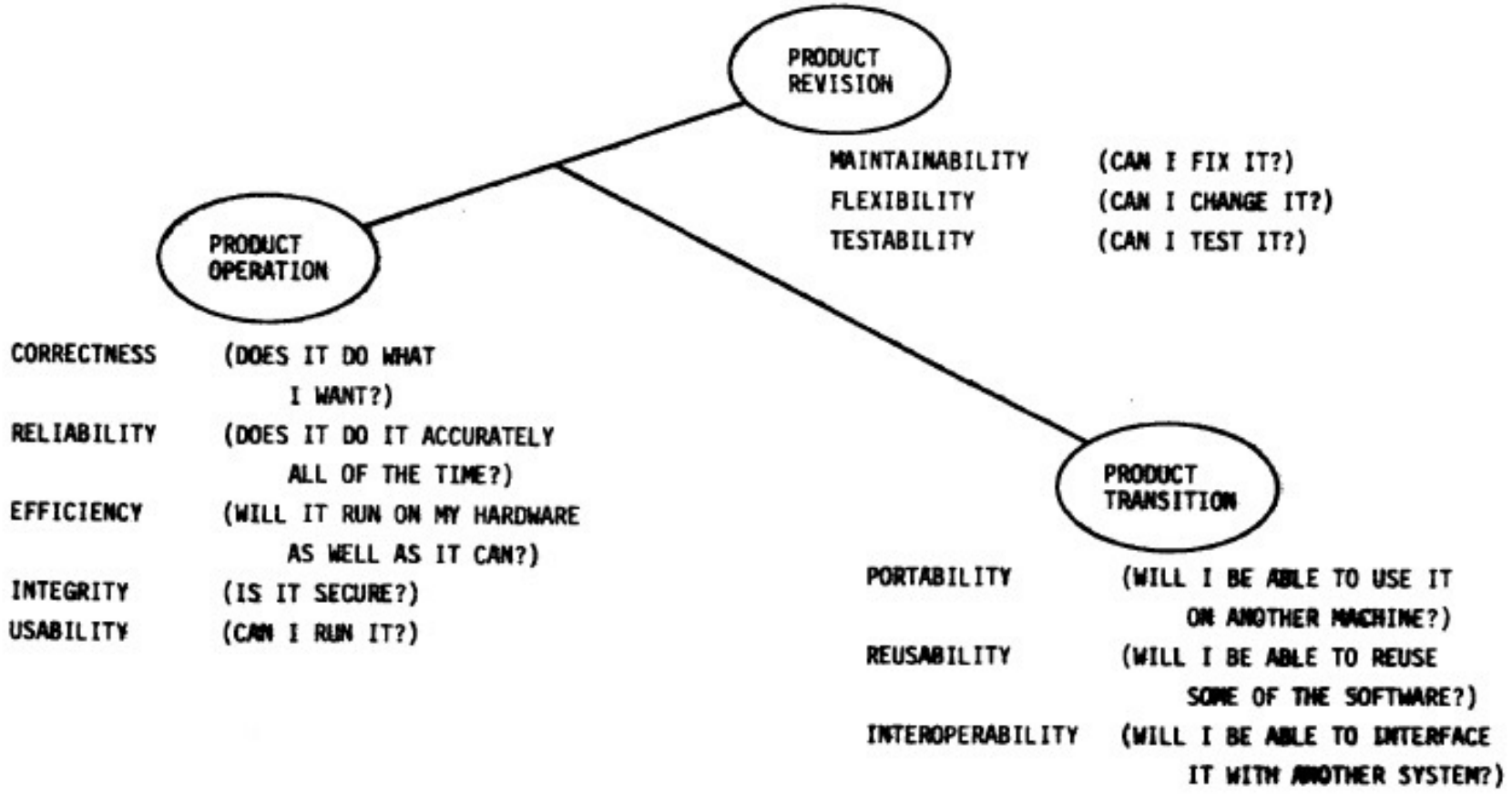


Figure 3.1-1 Allocation of Software Quality Factors to Product Activity 1352

Case study - Correctness

- Definition: *Extent to which a program satisfies its specifications and fulfills the user's mission objectives.*
- Life-cycle involvement:
  - Measured in development: analysis, design, implementation
  - Impact realized in: testing, operation, maintenance
- Criteria:
  - Traceability
  - Consistency
  - Completeness
- Criteria definition
- Metrics & criteria evaluation

- **Traceability**: Those attributes of the software that provide a thread from the requirements to the implementation with respect to the specific development and operational environment.
- **Consistency**: Those attributes of the software that provide uniform design and implementation techniques and notation.
  - Reliability + Maintainability
- **Completeness**: Those attributes of the software that provide full implementation of the functions required.

Metrics for criteria

- McCall.pdf:
  - pg 64 – Computation
  - pg 90 - Explanation

- Project assignment:
  - 1st step: choose your SQ model

    + choose factors (argue why?)