앞서 5장까지는 개발자의 입장에서 바라봄

지금부터는 프로젝트 관리자의 입장에서 바라봄

Unit 6.
ISO 9126 Quality Model

Organization of ISO 9126

- **Part 1**
  - Quality Model
- **Part 2**
  - External Metrics
- **Part 3**
  - Internal Metrics
- **Part 4**
  - Quality In Use Metrics

소프트웨어는 정량적으로 평가 할 수 없다

그렇지만 우리는 지표가 필요하기 때문에 iso에서
s/w 퀄리티를 정량적으로 평가하기 위한 지표를 줌

수학적인 식을 이용하여!

Quality Model Framework

- **Process Quality**
  - Quality of Life-cycle Process
  - Process quality contributes to improving product quality.
- **Product Quality**
  - Can be evaluated by measuring internal attributes or measuring external attributes.
  - Internal quality
    - is evaluated by the static measure of intermediate products.
    - View at Technical Level
  - External quality
    - is evaluated by measuring the behavior of the code when executed.
    - View of User/Management
  - Product quality contributes to improving quality in use.

우리가 지금까지 배운
diagram이나 discription 등 모든
중간 산출물

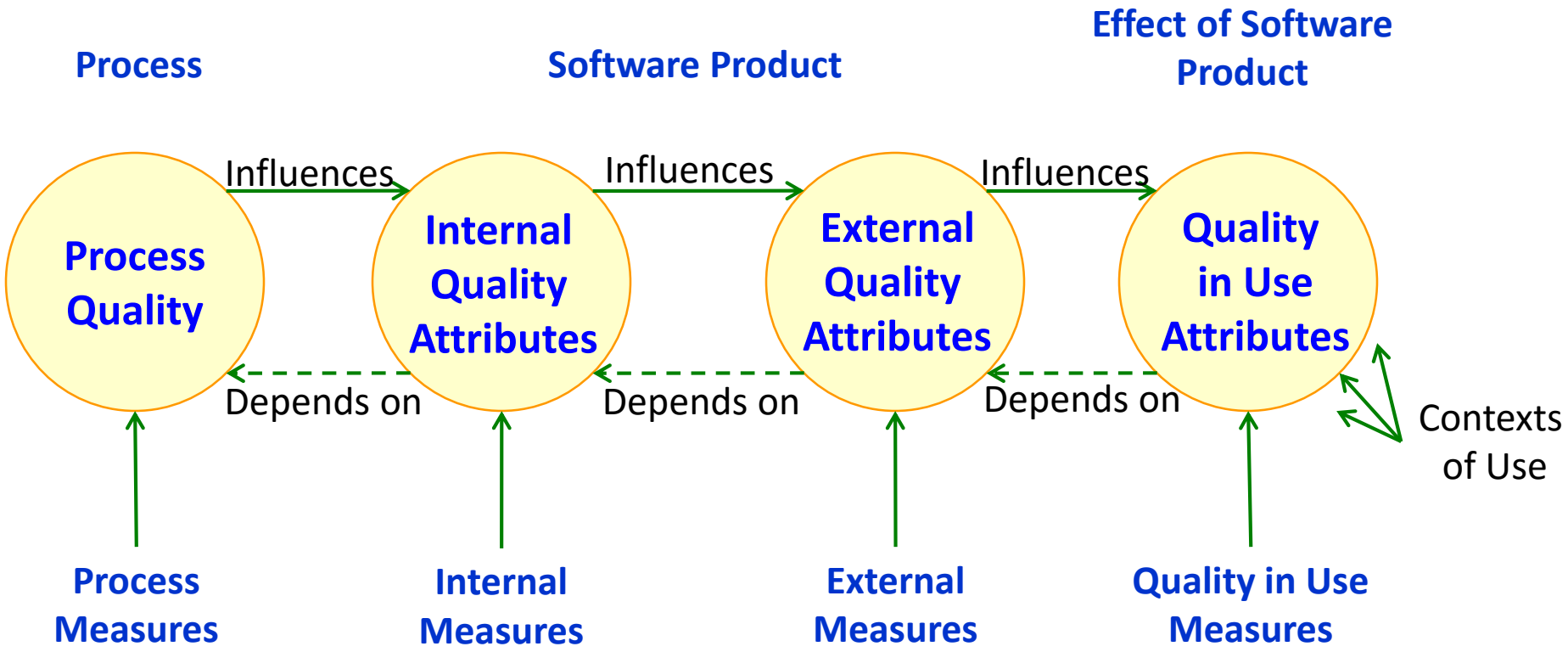
Quality Model Framework

- **Quality In Use**
  - User's view of the quality of an environment containing software, and is measured from the results of using the software in the environment.
    - Rather than properties of the software itself.
  - User's environment may be different from development environment.

사용자 환경에서의 소프트웨어 퀄리티를 확인하게 되는부분!

Quality Model Framework

- **Quality in the Lifecycle**

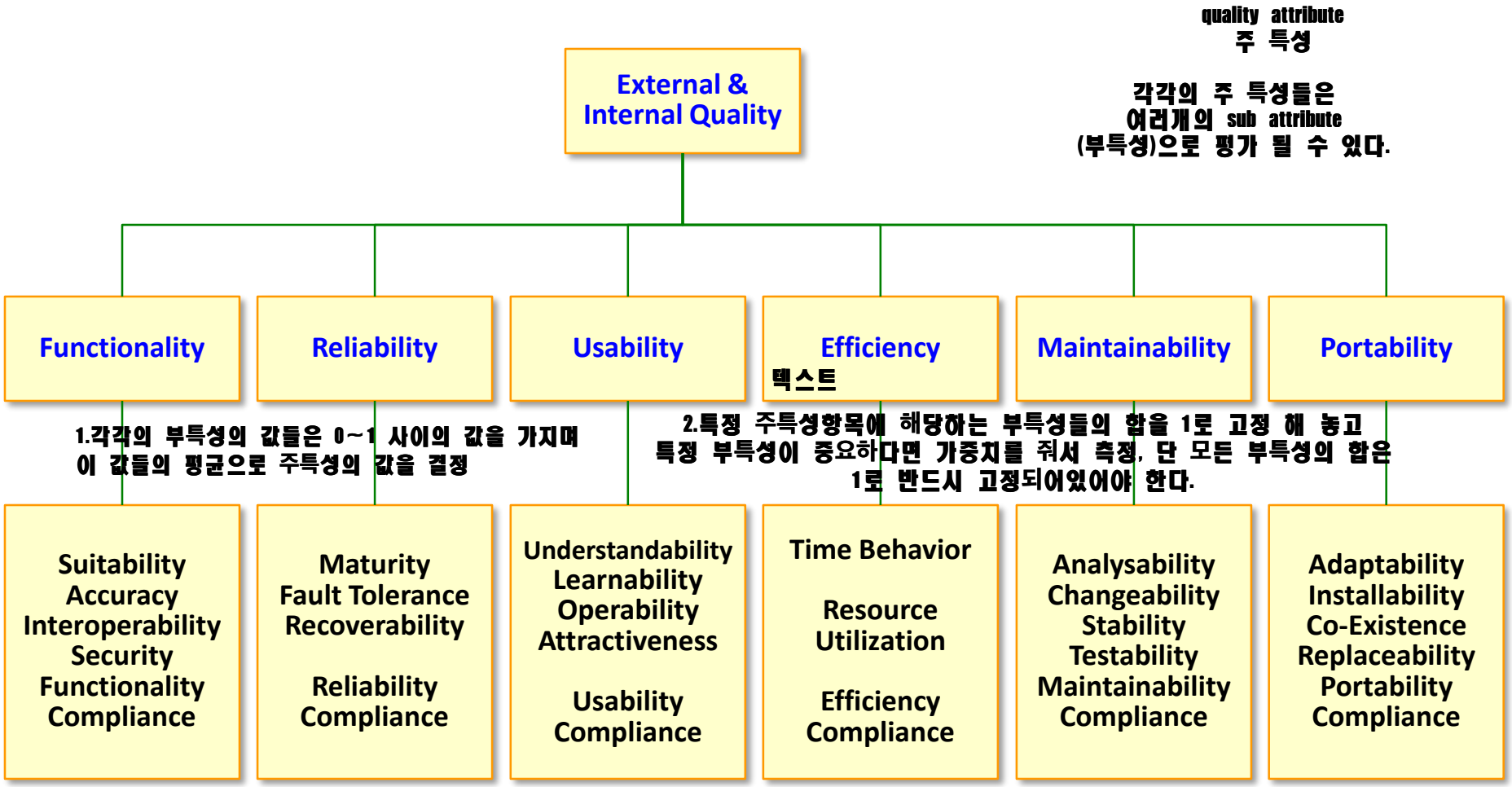


Quality Model of ISO 9126

- **Level 1.**
  - Characteristic
- **Level 2.**
  - Sub-characteristic
- **Level 3.**
  - Metrics
    - Formula

Quality Model of ISO 9126

- **Quality Model for External and Internal Quality**



Six Characteristics

- **1. Functionality**
  - Capability of software product to provide functions which meet stated and implied needs
- **2. Reliability**
  - Capability of software product to maintain a specified level of performance
- **3. Usability**
  - Capability of software product to be understood, learned, used and attractive to user

Six Characteristics

- **4. Efficiency** 얼마나 빨리 결과를 내고, 자원을 얼마나 사용하는가 즉, 공간/시간을 얼마나 차지하는가
  - Capability of software product to provide appropriate performance, relative to the amount of resources used
- **5. Maintainability**
  - Capability of software product to be modified
- **6. Portability** 내가 c로 짯더라도, 이식성이 좋으면 java 등과 같은 다른언어로 변환이 쉬움
  - Capability of software product to be transferred from one environment to another

skip

Sub-characteristics of Functionality

- **Suitability**
  - Capability of the software product to provide an appropriate set of functions for specified tasks and user objectives
- **Accuracy**
  - Capability of the software product to provide the right or agreed results or effects with the needed degree of precision
- **Interoperability**
  - Capability of the software product to interact with one or more specified systems

Sub-characteristics of Functionality

- **Security**
  - Capability of the software product to protect information and data so that unauthorised persons or systems cannot read or modify them and authorised persons or systems are not denied access to them
- **Functionality Compliance**
  - Capability of the software product to adhere to standards, conventions or regulations in laws and similar prescriptions relating to functionality

Sub-characteristics of Reliability

- **Maturity** 성숙성
  - Capability of software product to avoid failure as a result of faults in the software

failure: 멈추는 것
fault:런타임에 일어나는 모든 문제

성숙성이 좋다는 것은
〉fault가 발생하더라도
failure를 발생시키지 않는것을 의미

- **Fault Tolerance**
  - Capability of software product to maintain a specified level of performance in cases of software faults or infringement of its specified interface

위배

중간에 fault가 발생하더라도 일정수준이상(요구사항에 정의된)의 performance를 보여주는 것
〉〉위에 maturity보다 구현이 더 어려움

Sub-characteristics of Reliability

- **Recoverability** failure가 발생했다고 할 지라도 ›› 이전의 상태로 빠르게 복원하는 능력
  - Capability of software product to re-establish a specified level of ~~level of~~ performance and recover the data directly affected in the case of a failure

- **Reliability Compliance** 신뢰준수성
  - Capability of software product to adhere to standards, conventions or regulations relating to reliability

Sub-characteristics of Usability

- **Understandability**
  - Capability of the software product to enable the user to understand whether the software is suitable, and how it can be used for particular tasks and conditions of use
- **Learnability**
  - Capability of the software product to enable the user to learn its application
- **Operability**
  - Capability of the software product to enable the user to operate and control it

Sub-characteristics of Usability

- **Attractiveness**
  - Capability of the software product to be attractive to the user
- **Usability Compliance**
  - Capability of the software product to adhere to standards, conventions, style guides or regulations relating to usability

Sub-characteristics of Efficiency

- **Time Behavior** 응답시간이 얼마나 빠른가, throughtput이 얼마나 좋은가
  - Capability of the software product to provide appropriate response and processing times and throughput rates when performing its function, under stated conditions
- **Resource Utilization** 보통 모바일의 경우 제한된 리소스를 사용해야 하므로 얼마나 적당하게 사용하는가(배터리, cpu등등)
  - Capability of the software product to use appropriate amounts and types of resources when the software performs its function under stated conditions
- **Efficiency Compliance**
  - Capability of the software product to adhere to standards or conventions relating to efficiency

Sub-characteristics of Maintainability

- **Analyzability**
  - Capability of the software product to be diagnosed for deficiencies or causes of failures in the software, or for the parts to be modified to be identified
- **Changeability**
  - Capability of the software product to enable a specified modification to be implemented
- **Stability**
  - Capability of the software product to avoid unexpected effects from modifications of the software

Sub-characteristics of Maintainability

- **Testability**
  - Capability of the software product to enable modified software to be validated
- **Maintainability Compliance**
  - Capability of the software product to adhere to standards or conventions relating to maintainability

Sub-characteristics of Portability

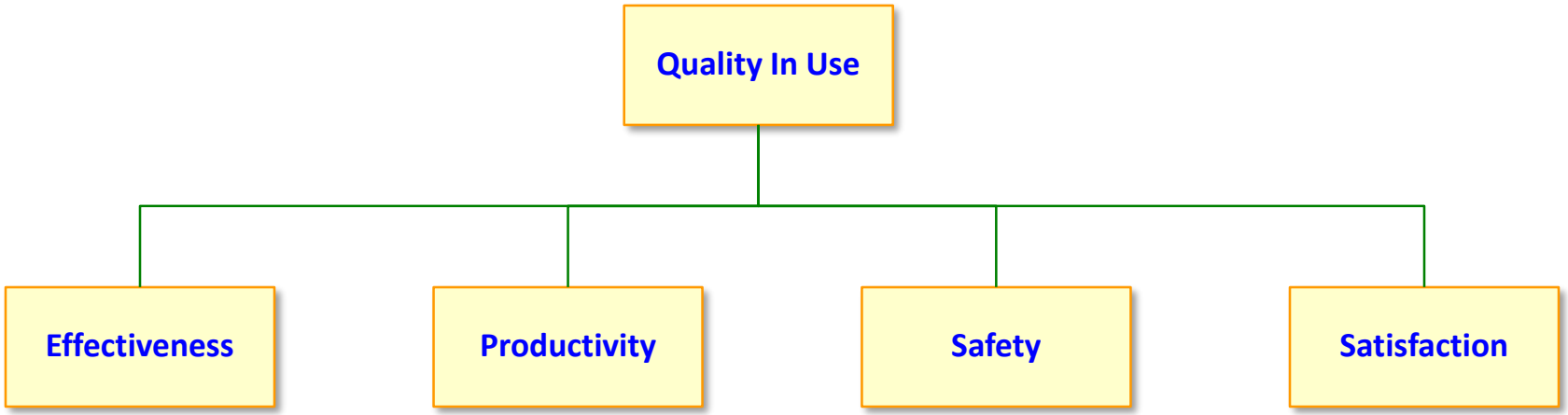
- **Adaptability**
  - Capability of the software product to be adapted for different specified environments without applying actions or means other than those provided for this purpose for the software considered
- **Installability**
  - Capability of the software product to be installed in a specified environment
- **Co-existence**
  - Capability of the software product to co-exist with other independent software in a common environment sharing common resources

Sub-characteristics of Portability

- **Replaceability**
  - Capability of the software product to be used in place of another specified software product for the same purpose in the same environment
- **Portable Compliance**
  - Capability of the software product to adhere to standards or conventions relating to portability

Quality In Use

- **Quality Model for Quality in Use**



Four Characteristics

- **1. Effectiveness**
  - Capability of the software product to enable users to achieve specified goals with accuracy and completeness in a specified context of use
- **2. Productivity**
  - Capability of the software product to enable users to expend appropriate amounts of resources in relation to the effectiveness achieved in a specified context of use

Four Characteristics

- **3. Safety**
  - Capability of the software product to achieve acceptable levels of risk of harm to people, business, software, property or the environment in a specified context of use
- **4. Satisfaction**
  - capability of the software product to satisfy users in a specified context of use