Software Quality

- **WHAT IS QUALITY?**
- **FOUR DIMENSIONS OF QUALITY**
  - ❖ **Specification quality**
  - ❖ **Design quality**
  - ❖ **Development (software construction) quality**
  - ❖ **Conformance quality**

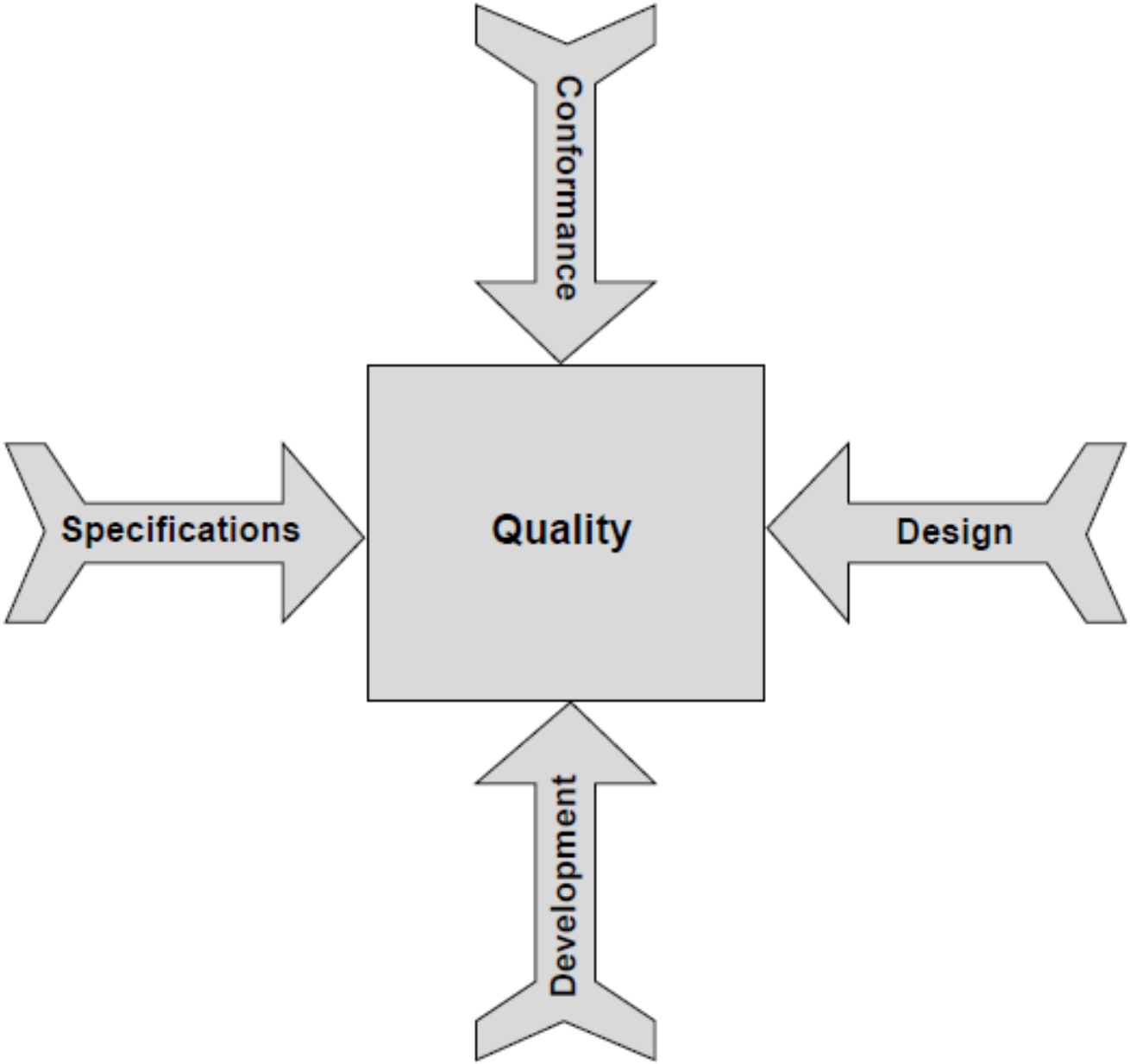
❖ WHAT IS QUALITY?

- To say that a certain product is a ***quality product*** implies that the product is of good quality
- On the other hand, people certainly use the term ***bad quality*** to express their dissatisfaction with the products or services they use
- Therefore, the adjective *good* is implicitly attached to the word *quality* in the minds of most people
- The International Organization for Standardization (ISO 9000, second edition, 2000) defines quality as the **degree** to which a set of inherent **characteristics** fulfills **requirements**

- Any product or service that meets the requirements of this definition is rated a “quality product/service”
- Any product or service that does not meet the requirements of this definition is rated “poor quality”
- ***Reliability*** of a product is its capability to function at the defined level of performance for the duration of its life

❖ FOUR DIMENSIONS OF QUALITY

- Quality has four dimensions
  - ❖**Specification quality**
  - ❖**Design quality**
  - ❖**Development (software construction) quality**
  - ❖**Conformance quality**
- Specifications are the starting point in the journey of providing a product or service, followed by design and then development
- Conformance quality is ensuring how well that quality is built into the deliverable at every stage


Conformance
Specifications
Quality
Design
Development

1) Specification Quality

- Specification quality refers to how well the specifications are defined for the product or service being provided
- Specifications have no predecessor activity, and all other activities succeed specifications
- Thus, if the specifications are weak, design will be weak, resulting in the development and manufacture of an incorrect product, and the effort spent on ensuring that quality is built in will have been wasted

- Specifications normally should include the following six aspects:

1) **Functionality aspects** : Specify what functions are to be achieved by the product or service
2) **Capacity aspects** : Specify the load the product can carry (such as 250 passengers on a plane or 100 concurrent users for a Web application)
3) **Intended use aspects** : Specify the need or needs the product or service satisfies
4) **Reliability aspects** : Specify how long the product can be enjoyed before it needs maintenance
5) **Safety aspects** : Specify the threshold levels for ensuring safety to persons and property from use of the product or service
6) **Security aspects** : Specify any threats for which the product or service needs to be prepared

- The tools for building quality into specifications are as follows:
  - **Process documentation** - Details the methodology for gathering, developing, analyzing, and finalizing the specifications
  - **Standards and guidelines, formats, and templates** - Specify the minimum set of specifications that needs to be built in
  - **Checklists** - Help analysts to ensure comprehensiveness of the specifications

2) Design Quality

- Design quality refers to how well the product or service to be delivered is designed. The objectives for design are to fulfill the specifications defined for the product or service being provided
- Therefore, if the design is weak, the product or service will fail, even if the specifications are very well defined
- In terms of software, conceptual design refers to software architecture, navigation, number of tiers, approaches to flexibility, portability, maintainability, and so on
- Engineering design refers to database design, program specifications, screen design, report design, etc.

- Software design normally contains the following elements:

1) Functionality design
2) Software architecture
3) Navigation
4) Database design
5) Development platform
6) Deployment platform
7) User interface design
8) Report design
9) Security
10) Fault tolerance
11) Capacity
12) Reliability
13) Maintainability
14) Efficiency and concurrence
15) Coupling and cohesion
16) Program specifications
17) Test design

- It is normal to conduct a ***brainstorming session*** at the beginning of a software design project, to select one optimum design alternative and to decide on the overall design aspects
- Such as the number of tiers, technology platform, software coupling and cohesion, etc.
- A brainstorming session helps designers arrive at the best possible solution for the project at hand

Ensuring Quality in Design

- Normally, software design is a two-step process:
  - **Conceptual design -** Referred to as high-level design, functional design specification, software requirements specification, and software architecture design
  - **Engineering design -** Referred to as low-level design, detailed design specification, software design description, and software program design

- The tools for building quality into design include the following:
  - **Process documentation** - Details the methodology for design alternatives to be considered, criteria for selecting the alternative for the project, and finalizing the conceptual design
  - **Standards and guidelines, formats, and templates** - Specify the possible software architectures along with their attendant advantages and disadvantages and so on
  - **Checklists** - Help designers to ensure that design is carried out comprehensively and appropriately

3) Development (software construction) Quality

The following activities form part of developing software:

- Create the database and table structures
- Develop dynamically linked libraries for common routines
- Develop screens
- Develop reports
- Develop unit test plans
- Develop associated process routines for all other aspects, such as security, efficiency, fault tolerance, etc.

- Good-quality construction is achieved by adhering to the ***coding guidelines*** of the programming language being used
- Normally there is a separate coding guideline for every programming language used in an organization
- Coding guidelines contain naming conventions, code formatting that help developers write reliable and defect-free code
- Of course, it is very important to have qualified people trained in software development
- Construction follows software design, and it should always conform to the design document
- In this way, good quality in construction can be achieved

- There are two techniques to ensure that quality is built into a product:
  - Reviews (walkthroughs)
  - Testing

4) Conformance Quality

- Conformance quality deals with how well an organization ensures that quality is built into a product through the above three dimensions
- It is one thing to do a quality job
- But it is quite another to unearth any defects lurking in the work product and ensure that a good-quality product is indeed built
- Essentially, conformance quality examines how well quality control is carried out in the organization

Ensuring Conformance Quality

- Ensuring that conformance quality is at desirable levels in the organization is achieved through :
  - Audits
  - Quality measurements
  - Metrics
  - Benchmarking
- Defect removal efficiency of verification and validation activities, defect injection rate, and defect density are all used for this purpose

- Audits also are conducted to ensure that projects conform to various applicable standards for building quality into all activities, including specifications and design
- In addition, organizational data is benchmarked against industry benchmarks, and corrective or preventive actions are taken to ensure that organizational conformance is indeed on a par with the industry
- Conformance quality is built in through process definition and continuous improvement for all software development activities as well as quality assurance

Quality dimension	How to build in quality	Techniques for ensuring quality
Quality of specifications	Specification development process documentation; standards and guidelines, formats, and templates for defining specifications; and checklists	Expert reviews, peer reviews, and brainstorming
Quality of design	Software design process documentation; standards and guidelines, formats, and templates for software design; and checklists	Expert reviews, peer reviews, managerial reviews, and brainstorming
Quality of development	Coding guidelines, configuration management, and change management	Peer reviews and software testing
Conformance quality	Diligent application of all quality assurance activities in the organization, process definition, and improvement	Audits, measurement and metrics for quality assurance activities, and benchmarking of organizational metrics against industry metrics