Quality Attributes

Meaning

A non-functional requirement and is not determined by function

Availability

Definition

- The probability that the system is available when a user uses 当用户使用系统时，系统可用的概率

- Maintenance shutdowns determined in advance are not included in the calculation 提前确定的停机维护不计入

Concerns

- Whether a failure occurred (unable to provide normal services, discovered by the outside world) 是否发生了故障（无法提供正常的服务，被外界发现）

- Consequences of failure 故障的后果

Tactics to improve

- Fault Detection
    - Ping/echo
    - Heartbeat（心跳）
    - Exceptions（异常）
- Fault Recovery
    - Vote（投票）
    - Active redundancy
    - Passive redundancy
    - Internal testing (Alpha version)
    - Checkpoint/rollback
- Fault Avoidance
    - Service offline
    - Transaction (事务)
    - Process monitoring

Modifiability

Concerns

- Cost of Modification

- Which parts of the system are modified

- When the modification occurs

- Who performs the modification

Tactics to improve

- Limiting the Scope of Modification
    - High cohesion, low coupling in modules
    - Consider potential modifications
    - Make modules generic
    - Hide information
    - Maintain a consistent interface
    - Limit communication paths
    - Use intermediaries
    - Name server
    - Create instances on demand
- Delaying Binding Time
    - Configuration files
    - Publish-subscribe style
    - Publish-subscribe style
    - Polymorphism
    - Concerns of Modifiability
    - Tactics to improve Modifiability

Perfermance

Concerns

- Speed of system response to events
- Related to the number and arrival pattern of events

Tactics to improve

- Resource requirements
    - Use more efficient algorithms
    - Reduce resource occupation when processing events
    - Control the rate of event arrival
    - Process only a subset of requests

- Limit execution time
    - Limit the length of the event queue to be processed
  - Resource management
    - Utilize concurrency mechanisms
    - Increase available resources
  - Resource arbitration
    - First-come, first-served
    - Fixed priority scheduling
    - Dynamic priority

Security

Concerns

- Resisting attacks on the system while ensuring legitimate users can use the system

Tactics to improve

- Resisting Attacks
  - User Authentication
  - User Authorization
  - Maintaining Data Confidentiality
  - Maintaining Data Integrity
  - Reducing Exposure
  - Access Control
- Detecting Attacks
  - Combination of Software and Humans
  - Identification of Attackers
- Recovering from Attacks
  - Restoring State

Testability

Concerns

- Make software's bugs easy to be tested
- Verify if the software product matches its requirement specifications (any discrepancies or omissions)
- Validate the quality of the software with the least cost and effort

Tactics to improve

- Black-box testing
  - Record / Replay
  - Separate interfaces from implementationss
  - Provide specific test paths
- White-box testing
  - Internal monitoring

Usability

Concerns

- Reduce the difficulty for users to use the software.

Tactics to improve

- Runtime Tactics
  - System Anticipates User Tasks
  - System Provides Appropriate Feedback to Users
  - System Provides Consistent Experience to Users
- Design-time Tactics
  - Support Undo Operations
  - Isolate the User Interface from Other Parts of the System