

## Quality Attributes

John Reekie  
University of Technology, Sydney

Contributors  
Lian Loke, University of Technology, Sydney

Terms of Use: Creative Commons Attribution-ShareAlike 2.5  
<http://creativecommons.org/licenses/by-sa/2.5/>

---

---

---

---

---

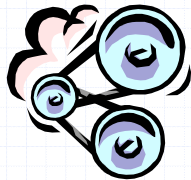
---

---

---

## Runtime qualities

- ◆ Runtime quality attributes emerge from the execution of a deployed system
- ◆ Quality scenarios relate to specific instances of runtime execution



---

---

---

---

---

---

---

---

## A useful acronym...

- p**erformance *Processing speed, resource utilization, response to load*
- v**sability *Human factors and impact*
- r**eliability *Failure rates, modes, severity, and recovery*
- s**ecurity *Data integrity, confidentiality, resistance to attack*

These quality attributes are a useful set of "umbrella" qualities.

---

---

---

---

---

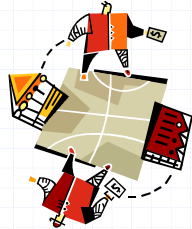
---

---

---

## Non-runtime qualities

- ◆ Non-runtime qualities relate to the lifetime of a system
- ◆ Quality scenarios are expressed in terms of incidents that occur during system development, deployment, or operation



---

---

---

---

---

---

---

---

## Acronym time again...

**M**aintainability

**E**volvability

**T**estability

**R**eusability

**I**ntegrity

**C**onfigurability

**S**calability

---

---

---

---

---

---

---

---

## Any more quality attributes?

*availability auditability modifiability feasibility  
compatibility backwards-compatibility standards-  
compliance continuity-of-view friendliness  
customizability learnability memorability enjoyability  
responsiveness schedulability verifiability  
analyzability reparability adaptability integrability  
interoperability predictability extensibility  
dependability safety portability survivability  
expendability expandability extensibility  
distributability flexibility*

---

---

---

---

---

---

---

---

## Performance

- ◆ Performance manifests in many ways:
  - Latency
  - Throughput
  - Memory efficiency
- ◆ Different performance metrics may apply to different parts of the system



---

---

---

---

---

---

---

---

## Usability

- ◆ Usability has many aspects:
  - Learnability
  - Enjoyability
  - Time to complete task
  - Error rates
- ◆ For good results, must be performed with a usability expert



---

---

---

---

---

---

---

---

## Reliability

- ◆ A complex field:
  - Hardware/software failures
  - Mean time to failure (MTTF)
  - Mean time to repair (MTTR)
- ◆ Reliability needs often depend on criticality:
  - Inconvenience
  - Loss of income
  - Loss of life

$$\text{Availability} = \frac{\text{MTTF}}{\text{MTTF} + \text{MTTR}}$$



---

---

---

---

---

---

---

---

## Security

- ◆ Almost any system can be under threat:
  - External attack (network)
  - Policy weakness
  - Data integrity
- ◆ High security is very expensive
- ◆ Secure expert help



---

---

---

---

---

---

---

---

## Firewalls

Packet filtering rules specify:

- ✦ Source address (or range)
- ✦ Destination address (or range)
- ✦ Destination port
- ✦ TCP/UDP
- ✦ Application protocol



Packet-filtering firewalls are a common and useful security measure



---

---

---

---

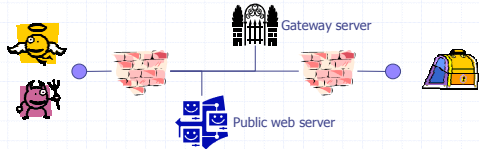
---

---

---

---

## Demilitarized zone (DMZ)



DMZ:

- ✦ Protects a whole enterprise, while
- ✦ Allowing specific services to be exposed, with
- ✦ Multiple security barriers

---

---

---

---

---

---

---

---

## Addressing quality attributes

- ◆ Quality narratives (or scenarios) → Coming right up!
- ◆ Behavior → Soon!
- ◆ Patterns → Later!
- ◆ Styles → In module 6!
- ◆ Tactics → See supplementary reading...

---

---

---

---

---

---

---

---

## Quality narratives

- ◆ A narrative or scenario that highlights a quality attribute need

- Context
- Action
- Response

He took his vorpal sword in hand:  
24 hours the manxome foe he sought  
So rested he by the Tumtum tree,  
And stood 5 minutes (mean) in thought.



Measurable if possible!

---

---

---

---

---

---

---

---

## Performance

"Under load at the peak event rate,  
any given event is analyzed and  
logged in no more than 200 ms."



---

---

---

---

---

---

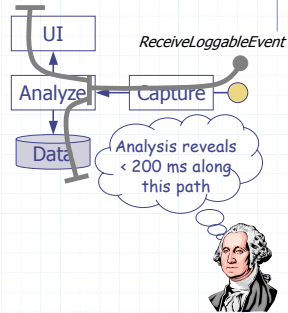
---

---

## Reasoning with use-case maps

◆ System quality scenarios can (often) be mapped to the architecture

- Reason about the behavior with respect to the desired quality



---

---

---

---

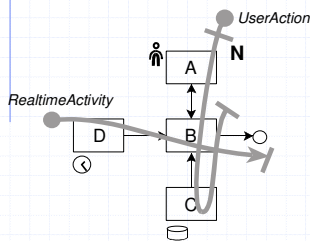
---

---

---

---

## Highlight performance issues



The use-case maps illustrate that component B is exercised by two scenarios with very different performance needs.

---

---

---

---

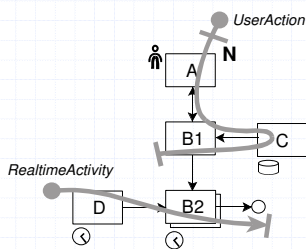
---

---

---

---

## Refactored



The component with a clash of performance requirements has been split into two. One has been replicated to allow for greater performance and scalability.

---

---

---

---

---

---

---

---

## Maintainability

"Two trained sysadmins can update the configuration files for a new multi-server configuration, and perform initial smoke tests, within eight billable hours."



---

---

---

---

---

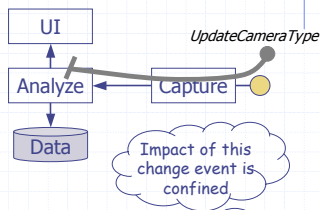
---

---

---

## Impact maps

- ◆ Architectural quality scenarios requires a new notation
  - Reason about impact on different elements of the architecture



---

---

---

---

---

---

---

---

## That's all, folks!

- ◆ Questions or comments?



---

---

---

---

---

---

---

---