Measures of System Quality: A 50-Year Evolution

John F. Meyer
jfm@umich.edu

IFIP WG 10.4
64th Meeting
Visegrad, Hungary
June 27-30,2013

Computer Science and Engineering
The University of Michigan

Outline

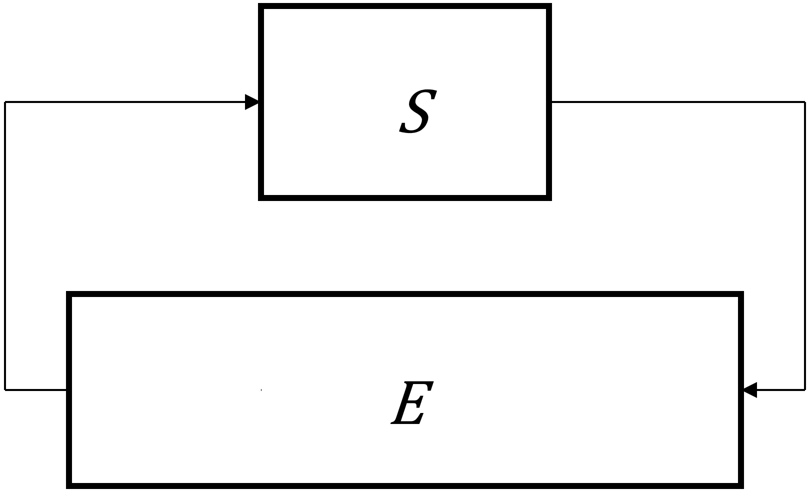
- Scope
    - When – Past half century
    - What
        - Measures of system quality
        - Fundamental modeling results
- Evolution
    - 1960s – Performance, reliability
    - 1970s – Degradable performance, performability
    - 1980s – Dependability, service quality (QoS)
    - 1990s – Other *X*ability, Qo*X* measures, security
    - 2000s – Subjective quality measures, resilience

Scope

- Measures of System Quality
- Quality:
  - A generic term with various interpretations
- System:
  - An IT system and its use environment
- Measures:
  - Probabilistic
  - Evaluation based on
    - system models (analytic, simulation, hybrid)
    - actual systems
    - combination of the two

A system and its use environment

- Let (S, E) denote the total system in question, consisting of an *object system* S and its *use environment* E.



- What S is or does in E can then be quantified via one or more quality measures.

More precisely …

- Generally, a quality measure can be viewed as a random variable Y_T, where
  - T is the *use period* during which the system is utilized or observed
    - syntactically an interval of some discrete or continuous *time base* I.
    - can range from a single instant $T = \{t\}$ to a period that's unbounded from above (long-run use)
  - Y_T takes values in a set of quality *outcomes*
- Probabilistic nature of Y_T is given by
  - partial descriptions: mean, higher order moments, selected probabilities
  - a full description: pdf (if it exists), PDF

Not just for analysis

- Although this abstraction appears to be specific to analytic models, it applies as well to
    - simulation models
    - actual systems
- In these cases, one obtains estimates of the probabilistic nature of Y_T, e.g., estimates of its
    - mean (expectation) $\mathrm{E}[Y_T]$
    - probabilities of the form $P[Y_T \leq y]$ or $P[Y_T = y]$ (if defined)

“What” versus “how”

- When discussing quality measures, it is helpful to distinguish
  - what property of S is being measured by Y_T

  from

  - how Y_T is formulated and evaluated (in terms of the dynamics of S and E).
- What: The name given to a measure typically suggests its meaning.
  - Generally, this is an interpretation of the values of Y_T (outcomes or probabilities thereof)
  - This is therefore a semantic issue, where specific meanings can vary according to the application.
- How: Described mainly by syntactic constructions; functions, equations, algorithms, distributions, etc.

Measure types

- Evaluations of computer and communication systems in the 1960s were principally concerned with two types of quality measures:
- Reliability: What a system is
  - Measures of the structural integrity of S in the presence of faults (independent of E).
  - Related to measures such as (structure-based) availability.
- Performance: What a system does
  - Measures of the effectiveness or efficiency of S in E, assuming both are fault-free.

Model assumptions: 1960s

- Reliability models (physical faults)
  - Structure of S (the representation thereof) is probabilistic
    - Dynamics are typically due to
      - rates of fault occurrences
      - durations of recovery actions
  - E is fixed (has a single state, representing constant active use of S)
- Performance models
  - Structure of S is fixed
  - E is probabilistic
    - Dynamics are typically due to
      - frequencies and durations of user demands (service requests)
      - workload imposed during active use

Structure-based measures

- Traditional structure-based measures of system reliability/availability convey a binary-valued view of a system's ability to serve its users:
  - Operational or up, meaning “capacity to serve"
  - Otherwise the system is non-operational or down
- Note that this dichotomy doesn’t necessarily coincide with what is experienced by a user in E (either a human or some other system).
- Indeed, a structure-based measure reflects what a user might experience only when S is constantly used by E.

Basic reliability/availability measures

- Relative to a continuous time base $I = \mathbb{R}_{\geq 0}$, let
  - $X(s)$ = 1 if S is up at time s; 0 else
  - $T = [0, t], \mathrm{t} > 0$

  and consider the quality measure
  - Y_T = the amount of time during T that S is operational
- Then $Y_T = \int_0^t X(s)ds$
- In turn, Y_T yields some other familiar quality measures:
  - Reliability (during T; down = failure): $P[Y_T = t]$
  - Interval availability: $\frac{1}{t} Y_T$
  - Limiting (steady-state) availability: $\lim_{t \to \infty} \frac{1}{t} Y_T$

Basic performance measures

- Relative to a continuous time base $I = \mathbb{R}_{\geq 0}$, let
  - $A(s)$ = # of job arrivals (service requests) from E during $[0, s]$
  - $L(s)$ = # of jobs in S at time s ($L(0) = 0$; no upper bound on $L(s)$)
- Then
  - $C(s)$ = # of job completions (to E) during $[0, s] = A(s) - L(s)$
- Accordingly, the throughput of S during T ($T = [0, t]$, $t > 0$) is the quality measure
  - $Y_T = C(t)/t$ (job completion rate)
- In the limit (which exists under appropriate conditions), the steady-state throughput is given by
  - $\lim_{t \to \infty} Y_T = \lim_{t \to \infty} (A(t) - L(t))/\mathrm{t} = \lim_{t \to \infty} A(t)/\mathrm{t} - \lim_{t \to \infty} L(t)/\mathrm{t} = \lim_{t \to \infty} A(t)/\mathrm{t}$,
    i.e., it coincides with the steady-state arrival rate.

Basic performance measures (cont'd)

- Some further measures of E and S:
- Let
  - $\alpha_T = A(t)\,/\mathrm{t} =$ job arrival rate from E during T
  - $L_T = \frac{1}{t}\int_0^t L(s)ds$ = time-average # of jobs in S during T
  - $W_T = \frac{1}{A(t)}\int_0^t L(s)ds$ = average time a job spends in S during T
  - *Then* with a "little" manipulation, we have
  - $L_T = \alpha_T W_T$

which, in terms of the limiting values (when they exist),

$$l = \lim_{t \to \infty} L_T \qquad \alpha = \lim_{t \to \infty} \alpha_T \qquad w = \lim_{t \to \infty} W_T$$

gives $l = \alpha\, w$ (Little's Theorem, 1961).

Fundamental results: 1960s

- Reliability
  - W.G. Bouricius, W.C. Carter and P.R. Schneider, "Reliability Modeling Techniques for Self-Repairing Computer Systems," in *Proceedings of the 24th ACM National Conference*, pp. 295-309, ACM, 1969.
    - Need to consider bounded use periods (missions) $T = [0, t]$
    - MTTF as $t \rightarrow \infty$ is a misleading measure for highly reliable systems (even when t is large)
    - Concept of coverage c
    - Sensitivity of reliability measures to values of c

Fundamental results: 1960s (cont'd)

- Performance
  - G. Estrin and L. Kleinrock, "Measures, Models and Measurements for Time-Shared Computer Utilities," in *Proceedings of the 22nd ACM National Conference*, pp. 85–96, ACM, 1967
    - Utility of queueing models for computer performance evaluation
    - Led to two classic books on *Queueing Systems* authored by Kleinrock and published in the mid-1970s:
      - Vol. 1: Theory
      - Vol. 2: Computer Applications

Be aware of the user: 1970s

- Appropriateness of an up-down, user-independent view of system reliability began to be questioned in the 1970s.
- This was due to developments in several areas:
  - Degradable computing systems
  - Computation-based measures
  - Studies examining the effects of workload on hardware reliability
  - Software reliability, software fault-tolerance
- Concerns with development faults in software were perhaps the most influential.
  - **A software system S, no matter how faulty, requires a non-trivial use environment E in order to fail.**

User-oriented measures

- Accordingly, more general types of quality measures began to emerge (mid-70s, early 80s), placing greater emphasis on how delivered services are affected by internal and external faults.
  - Performability: Measures of a system's ability to perform (serve its users) throughout a specified utilization period.
  - Dependability: Measures of a system's trustworthiness with respect to delivery of a specified service.
  - QoS: The "collective effect" of service performances (including dependability) which determine the degree of satisfaction of a user of the service (telecom ITU-T Recommendation E.800).

“Does” trumps “is”: 1980s

- As a consequence of the concept and terminology efforts of this Working Group led by Laprie in the early 80s:
  - Summer 1981 WG meeting devoted to this subject
  - Panel of terminology papers at FTCS-12
  - Laprie paper presented at FTCS-15

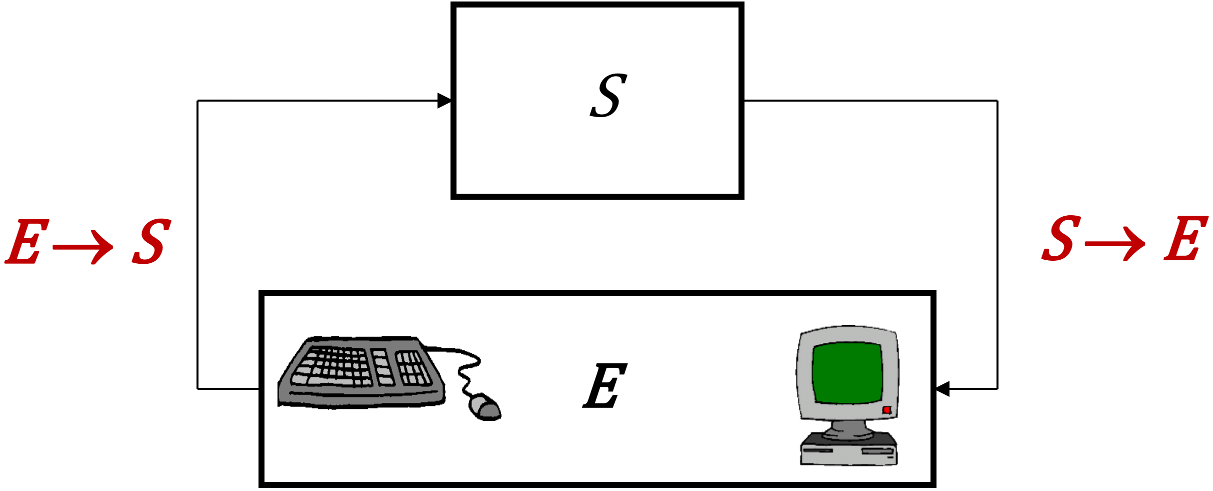
  a distinguishing feature of dependability was its treatment of the notion of “failure.”
- Instead of it being a loss of capacity to serve (per traditional measures of reliability and availability), a (service) *failure* is identified with a transition from correct to incorrect service delivery.

Why was this important?

- Dissemination of this view during the 1980s and early 90s produced a major change in how various dependability attributes (particularly reliability and availability) were measured and evaluated.
- It anticipated the emergence of user-centric applications.
  - Personal computing
  - Embedded computers in home appliances, entertainment systems, cars, trains, aircraft, …
  - Home networks, enterprise networks, ATC systems, military C2 systems
  - World-wide communication and information sharing

The use environment E

- When quantifying system quality from a user's perspective, two important aspects of E need to be considered:



- $E \rightarrow S$: User demands, other influences external to S
- $S \rightarrow E$: Services delivered by S

Some examples of $E \rightarrow S$

- User demands
  - Workload (computer systems)
  - Call, message, and connection traffic (communication networks)
- External faults
  - Radiation
  - Electromagnetic interference
  - Cyber attacks
- Unanticipated environmental changes of the type tolerated by resilient systems
- Generally, the dynamics of the above can be described objectively in technical terms.

$S \to E$

- This is mainly where system quality is observed by users.
- With respect to measure types that account for effects of faults originating in both S and E:
  - Dependability: Quality of S to the extent that services are delivered properly to E (failures occurs when they are not).
  - Performability: Quality of services delivered throughout a specified use period T (perhaps unbounded).
  - QoS: The “collective effect” of service performances (including dependability) which determine the degree of satisfaction of a user of the service.

Measures of security: 1990s

- What is and is not common between dependability and security was first discussed seriously at a joint D-S WG (10.4, 11.3) workshop held at the Grand Canyon, AZ in 1991.
- This mutual interest has continued since then, so what about (quantitative, probabilistic) quality measures in this regard?

Confidentiality

- Relative to the well-known security “trifecta” (CIA Triad)
  - Confidentiality
  - Integrity
  - Availability

  measures of I and A have been understood and used by the dependability community for many years.
- So what remains w.r.t. the Triad are measures of confidentiality.

Confidentiality measures

- To illustrate what such a measure might look like according to the ground rules of this review (it is quantitative and probabilistic), let
  - BoC denote a breach of confidentiality (where a BOC can be any one of several events that constitute unauthorized access to or disclosure of confidential information in S).
- Relative to a continuous time base $I = \mathbb{R}_{\geq 0}$, let
  - $B(s) =$ # of BoC-occurrences in S during $[0, s]$.
- Then for $T = [0, t]$, $t > 0$
  - $Y_T = B(t) =$ # of BoC-occurrences in S during T.
- What's missing in this picture?

User-perceived quality: 2000s

- Although the wording of the ITU-T QoS definition suggests that it is somewhat subjective, e.g., phrases such as
  - “collective effect”
  - “degree of satisfaction”

  practical use of this term has not been.
- Consequently, more explicitly subjective concepts of quality emerged during the 2000s.
  - Quality of experience (QoE):
    - The overall acceptability of an application or service, as perceived subjectively by the end-user (ITU-T Study Group 12, Geneva, January 2007)
  - Quality of perception (QoP):
    - End-user perception (as in QoE) along with an understanding and assimilation of what is perceived.

Resilience

- During the past decade, system resilience has received increased attention in several system domains.
  - Internet
    - IRIS (Infrastructure for Resilient Internet Systems)
  - Information system technology
    - ReSIST (Resilience for Survivability in IST)
  - High Performance Computing (HPC)
    - Resilience in HPC
  - Safety systems
    - Resilience engineering
  - Industrial, ecological, and social systems
    - Ohio State University’s Center for Resilience
  - Defense systems
    - US DoD initiative: Engineered Resilient Systems

ReSIST definition

- Quoting from Jean-Claude Laprie's 2008 DSN paper (Alaska):

> With such ubiquitous systems, what is at stake is to maintain dependability, i.e., the ability to deliver service that can justifiably be trusted in spite of continuous changes. Our definition of resilience is then:
>
> > The persistence of service delivery that can be justifiably be trusted, when facing changes.
>
> The definition given above builds on the initial definition of dependability, which emphasizes justifiably trusted service.

Shorthand versions

- ReSIST:
  - *Def.*: *Resilience* is the persistence of dependability when facing changes.
- Extending the definition to account for properties such as degradable performance:
  - *Def.*: *Resilience* is the persistence of performability when facing changes.

Resilience measures

- Per the shorthand definitions just cited, resilience measures are really nothing new.
- For example, given an object system S and use environment E, one can select a favorite quantitative x-measure (x = dependability or performability) and then specify what is meant by persistence.
- For example, if “to persist" is “to exist" then the resulting resilience measure coincides with the underlying x-measure, except it can now reflect effects of changes (including faults).

Resilience measures (cont'd)

- More restricted interpretations of "persist" correspond to more specialized measures of resilience.
- For example, suppose "persist" has the stronger meaning of "holding on" to some acceptable level of ability to serve during the use period T, e.g.,
  - stay at or above some lower bound $\boldsymbol{b}$ on the mean service quality (MSQ) throughout T
- Resilience is then quantified by the performability measure:
  - Y_T = fraction of T wherein MSQ $\geq$ $\boldsymbol{b}$.

Needs looking ahead

- More refined dependability/performability measures for contemporary systems ranging from
  - Embedded

  to

  - Ubiquitous (cloud is a component)
- Measures of system security whose probabilistic nature can be evaluated by practical means based on
  - Models
  - 7Experiments
  - Field data