

Software Reliability and Safety

CSE 8317 — Spring 2015

Prof. Jeff Tian, tian@engr.smu.edu
CSE, SMU, Dallas, TX 75275
(214) 768-2861; Fax: (214) 768-3085
www.engr.smu.edu/~tian/class/8317.15s

OV. Overview

- Quality/Dependability, Reliability, and Safety
- SRE: Software Reliability Engineering
- SSE: Software Safety Engineering
- CSE 8317 Perspective and Common Analyses

Quality and Dependability

- ISO 9126 quality characteristics:
 - ▷ functionality, reliability, usability, efficiency, maintainability, portability
 - ▷ Characteristics into sub-characteristics (strict hierarchy)
 - ▷ customized for companies
 - e.g., IBM's CUPRIMDSO.
 - ▷ adapted to application domains
 - reliability, usability, security for Web

- Dependability: “The trustworthiness of a computing system which allows reliance to be justifiably placed on the services it delivers” (IFIP WG10.4).
 - ▷ reliability, availability, safety, security.
 - ▷ integrity and maintainability (?)
 - ▷ security sub-attributes:
 - availability, confidentiality, integrity

Quality: Views and Aspects

View	Attribute	
	Correctness	Effectiveness
Customer (external)	Failures: reliability safety	Usability Maintainability Portability Performance Installability Readability
Developer (internal)	Faults: count distr class hazard	Design Size Change Complexity control data presentation

- 8317: Reliability/safety focus
- Things affect reliability/safety

What Is Reliability?

- *Reliability*: Probability of failure-free operation for a specific time period or for a given set of input conditions under a specific environment
 - ▷ Probability: quantitative/statistical
 - ▷ Failure: behavioral deviations
 - ▷ Time vs. input measurement/sampling
 - ▷ Environment: OP and UBST

- Software reliability engineering (SRE):
 - ▷ Failure and other measurement/data
 - ▷ Reliability assessment
 - ▷ Reliability and other predictions
 - ▷ Decision making and management
 - ▷ Reliability and process improvement

What Is Safety?

- *Safety*: The property of being accident-free for (embedded) software systems.
 - ▷ Accident: failures with severe consequences
 - ▷ Hazard: condition for accident
 - ▷ Related to but distinct from reliability
 - ▷ Specialized techniques

- Software safety engineering (SSE):
 - ▷ Failure prevention and fault tolerance
 - ▷ Hazard identification/analysis techniques
 - ▷ Hazard resolution alternatives
 - ▷ Safety and risk assessment
 - ▷ Qualitative focus
 - ▷ Safety and process improvement

Reliability, Safety and Defects

- Reliability/safety negatively (and directly) correlated to defect (failure view).

- Defect/bug definition: SQE Ch.2
 - ▷ Failure: external behavior
 - deviation from expected behavior
 - ▷ Fault: internal characteristics
 - cause for failures
 - ▷ Error: missing/incorrect actions
 - ▷ Causal relation, but not necessarily 1-1
 - ▷ Safety-related: accident & hazard

- Defect and quality assurance: SQE Ch.3
 - ▷ Preventive actions based on analysis
 - ▷ Fault (detection &) removal: insp./testing/etc.
 - ▷ Fault tolerance (and safety assurance)

QA for Reliability/Safety Assurance

- Defect prevention:
 - ▷ Error source elimination
 - ▷ Error blocking

- Defect removal: Inspection/testing/etc.

- Defect tolerance:
 - ▷ Fault tolerance (failure↓)
 - ▷ Damage minimization (safety)

- Link to reliability/safety
 - ▷ All help assure reliability/safety
 - ▷ SQE/slides online

QA for Reliability/Safety Assurance

- SRE relation/applications:
 - ▷ Functional relation: reliability \sim failure
 - ▷ QA alternatives directly work with SRE
 - ▷ QA affects results/failures via causal chain
error \Rightarrow fault \Rightarrow failure
 - ▷ Closer to failure
 \Rightarrow closer to SRE activities
(e.g., system and acceptance testing)
- SSE relation/applications:
 - ▷ More focused (not as broad)
 - ▷ Hazard focus (small subset of failures)
 - ▷ SSP: QA throughout dev. process
- Specifics to be examined later

QA for Reliability/Safety Assurance

- Inspection:
 - ▷ Wide applicability (diff periods/artifacts)
 - ▷ Conceptual/static faults
 - ▷ Human intensive, varied cost

- Applications in SRE and SSE
 - ▷ Fault eliminations:
 - helps both reliability and safety
 - SRE/SSE \sim high/low fault densities
 - ▷ Scenario-based (focused) inspection:
 - SRE: common usage
 - SSE: FTA/ETA-based
 - ▷ Early reliability prediction
 - ▷ Safety constraints and inspection

QA for Reliability/Safety Assurance

- Formal verification: SQE Ch.15
 - ▷ Works on code with formal spec.
 - ▷ Practicality: high cost → benefit?
 - ▷ Human intensive, rigorous training

- Applications in SRE and SSE
 - ▷ High cost ⇒ mostly in SSE
 - ▷ Module SSE.3
 - ▷ Focus through FTA and/or ETA
 - ▷ Leveson's approach:
 - safety and other constraints
 - carried through dev. process
 - ▷ Other adaptations:
 - table-driven, model checking, etc
 - PSC, module SSE.4

QA for Reliability/Safety Assurance

- Testing:
 - ▷ Dynamic/run-time/interaction problems
 - ▷ BBT/WBT: external vs internal focus
 - ▷ Coverage/usage: termination criteria

- Applications in SRE and SSE
 - ▷ Chief application domain for SRE
 - ▷ OP-based testing (UBST):
 - basis for reliability modeling
 - ▷ Earlier phases:
 - WBT/BBT with coverage
 - ▷ Indirect link to SSE

QA for Reliability/Safety Assurance

- Fault tolerance:
 - ▷ Dynamic problems
 - ▷ Technique problems (independent NVP?)
 - ▷ Process/technology intensive
 - ▷ High cost

- Applications in SRE and SSE
 - ▷ Too expensive for regular SRE
 - ▷ As hazard reduction/control in SSE
 - ▷ Other related SSE techniques:
 - general redundancy
 - substitution/choice of modules
 - barriers and locks
 - analysis of FT

Measurement, Analysis, & Modeling

- Measurements: SQE Ch.18
 - ▷ Result: success/failure/accident/etc.
 - ▷ Indirect measurements, as predictors:
 - activity/product internal/environment

- Analysis and modeling:
 - ▷ Model categories/context: SQE Ch.19
 - ▷ Defect analysis: SQE Ch.20
 - ▷ Risk identification: SQE Ch.21
 - ▷ Common basis for SRE & SSE
 - ▷ SRE/SSE models:
 - Data ⇒ reliability & safety

- 8317 focus: Analysis-based resolution for reliability/safety assurance and improvement

Reliability Analyses and Models

- SRE.2/3: model = function relations
e.g., failure \sim time or input.

- Time domain approach
 - ▷ Failure arrival process
 - ▷ Statistical modeling
 - ▷ Failure count/interval/rate data
 - ▷ Time and other measurements
 - ▷ SRGMs: s/w reliability growth models
 - ▷ Assessment/prediction/decisions

- Input domain approach
 - ▷ Repeated random sampling
 - ▷ Related definitions and models
 - input domain reliability models
 - ▷ Fault seeding models

Reliability Analyses and Models

- TBRMs: tree-based reliability models
 - ▷ Both time/input domain info.
 - ▷ Additional benefit:
 - risk identification
 - guide for focused remedial actions
 - ▷ Technique: tree-based modeling
 - ▷ Development/application/SMU research
 - ▷ Major focus in 8317 (SRE.2)

- Other related issues: SRE.4
 - ▷ Implementation & applications
 - ▷ OP development & QA activities
 - ▷ Fault/defect modeling
 - ▷ Data treatment
 - ▷ Reliability composition, etc.

Safety Analysis & Improvement

- Hazard analysis and resolution (SSE.2)
 - ▷ Focus: accidents and pre-conditions (hazards), not other failures
 - ▷ “Safeware” Ch.13-16 & SQE Ch. 16.4
 - ▷ Identification and analysis
 - ▷ Resolution: elimination/reduction/control
 - ▷ Integration in development process
 - SSP (software safety program)
 - “Safeware”, Part IV (Ch.11-18)

- Formal verification related:
 - ▷ Main part: SSE.3, SQE Ch. 15.
 - ▷ PSC: SSE.4, SQE Ch. 16.5

Safety Analysis & Improvement

- Hazard analysis:
 - ▷ Fault trees: (static) logical conditions
 - ▷ Event trees: dynamic sequences
 - ▷ Other analyses
 - ▷ Generally qualitative
 - ▷ Related: hazard and risk assessment

- Hazard resolution (pre-accident)
 - ▷ Negate/block/mitigate/etc.
 - ▷ Hazard elimination/reduction/control

- Related: damage reduction (post-accident)

Safety Assurance & Improvement

- **Eliminate** identified hazard sources in material/component/software/etc.

- **Reduce** hazard likelihood/severity via:
 - ▷ Creating hazard barriers,
 - ▷ Minimizing failure probability, etc.

- **Control** hazard (after detection) via:
 - ▷ Isolation and containment,
 - ▷ Fail-safe design, etc.

- **Reduce** damage (post-accident, as compared to pre-accident for the above)

How CSE 8317 Fits In?

- Software reliability engineering (SRE):
 - ▷ SRGMs/IDRMs: assessment/prediction;
 - ▷ TBRMs and other recent development;
 - ▷ Focus: reliability analysis/improvement.

- Software safety engineering (SSE):
 - ▷ Fault/event tree analyses, etc.;
 - ▷ Hazard elimination/reduction/control;
 - ▷ Process integration, FV, FT, PSC, etc.

- Common analyses/techniques:
 - ▷ defect analysis (SQE Ch.20)
 - ▷ risk identification: SQE Ch.21