Panel on “Software Testing & Evaluation” and “Dependability Benchmarking”

WG 10.4 meeting, Paraty
Goal of the Panel

“to discuss contact points between software testing & evaluation and dependability benchmarking, and find possible contribution from these well established areas to dependability benchmarking”

Quotation from our workshop organisers
The panellists

- Karama Kanoun, LAAS-CNRS, France: “Dependability Benchmarker and Software Evaluator”
- Mladen Vouk, North Carolina State University, Raleigh, USA: “Software Tester and Software Evaluator”
- Marie-Claude Gaudel, LRI, Université de Paris-Sud, France: “Software Tester and ... chair of the panel” 😊
Question 1, from our organisers

- If a dependability benchmark is a test (or set of tests) to quantify computer dependability, what is the overlap of this with software testing and software evaluation?
  - Comment 1: “computerised systems dependability” rather than “computer dependability”?
  - Comment 2: benchmarks need to be specifically designed to be representative of field usage... as is software evaluation, but not always testing
Questions 2 & 3

- What tests are out there for software characteristics other than functional correctness (i.e. software reliability, safety, security, etc)?
  - Personal (and biased) comment: in order to do testing in a meaningful way, it is necessary to have requirements provided in an unambiguous, verifiable way

- Is the notion of software reliability the same for the software testing community and for the dependability community?
Questions 4 & 5

- Is the software reliability a feature of the software (i.e. a given program) or it also depends on the environment (the HW platform, the OS and middleware, system workload, operational profile, etc)?

- What are the possible contributions of
  - mutation,
  - software metrics,
  - formal methods, etc
  to dependability benchmarking?
Let us listen to the panellists
A testing point of view (1)

- *What tests are out there for software characteristics other than functional correctness (i.e. software reliability, safety, security, etc)?*

- Few methods…
  - Operational profile => reliability testing,
  - Elayne Weyuker’s works on performance testing,
  - Testing driven by objectives => safety testing
  - … more inputs from the audience ?

- Role of Fault Injection ?
  - Robustness testing more than reliability testing…
A testing point of view (2)

- Is the notion of software reliability the same for “the” software testing community and for the dependability community?

- Which software testing community? 😊
  Probably yes

- Is the software reliability a feature of the software (i.e. a given program) or it also depends on the environment (the HW platform, the OS and middleware, system workload, operational profile, etc)?

- A program is just a piece of text… cf. [Galves & Gaudel 98] on the failures of a correct program
The testing point of view (3)

- Test sets are almost never generic (exception: compiler certification; I feel able to design a reasonable dependability benchmark for a compiler, or at least I hope so 😊)
- Benchmark = standard test set, or standard test procedure?
- For “usual” testing, the choice of a test strategy is strongly dependent on
  - Controllability issues
  - Observability issues
  - Quality requirements, and among them, reliability issues, security issues, etc