Fundamentals of Testing

Software Reliability and Testing - Barbara Russo

SwSE - Software and Systems Engineering research group

unibz

Freie Universität Bozen
Libera Università di Bolzano
Università Liedia de Bulsan

Goal of testing

Testing is the process of executing a program with the intent of finding errors

Glen Myers

Testing

Software testing is the process of analysing a software item to detect the differences between ***existing and required conditions*** *(i.e., bugs) and to evaluate the features of the software item*

IEEE definition

Limits of software testing

> *Program testing can be used to show the presence of bugs, but never to show their absence!*
>
> Dijkstra, 1969

Limits of software testing

- If a failure is detected then the software is a failure software, but
- If no failure has been detected we cannot say that the software is correct
- **Exhaustive testing is not feasible**, but we can compute the probability that no failures occur in a given interval of time

—> **software reliability**

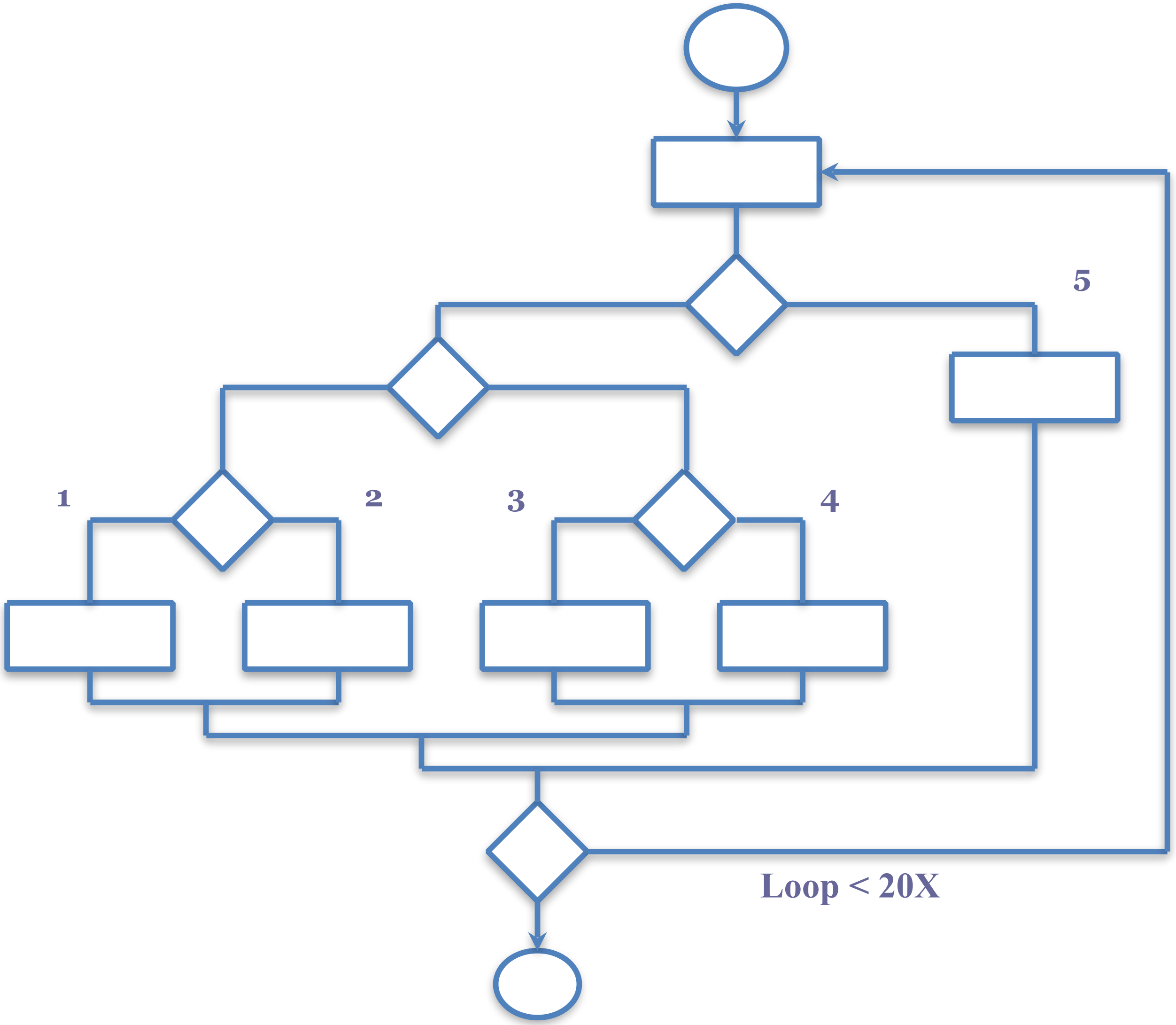
Limits of software testing

Beware of bugs in the above code; I have only proved it correct, not tried it

Knuth, 1977

- We need to test for confidence not for proof of correctness

Exhaustive testing



if we execute one test per millisecond, it would take too much to test this program!!

The Peculiarity of Testing

- Testing is the process of executing a program with the intent of finding an error
- A good test case has a high probability of finding an as-yet undiscovered error
- A successful test case is one that uncovers an as-yet undiscovered error

Basic questions

- When does testing start? When does it complete?
- What techniques should be applied during software development to get acceptable quality at acceptable cost?
- How can we assess the readiness of a product to release?
- How can we control the quality of a product to release?

When does testing start? When does it complete?

- Testing activities are spread all over the development process
- There are several techniques at different stages of the process
- Different techniques involve different stakeholders

When does testing start? When does it complete?

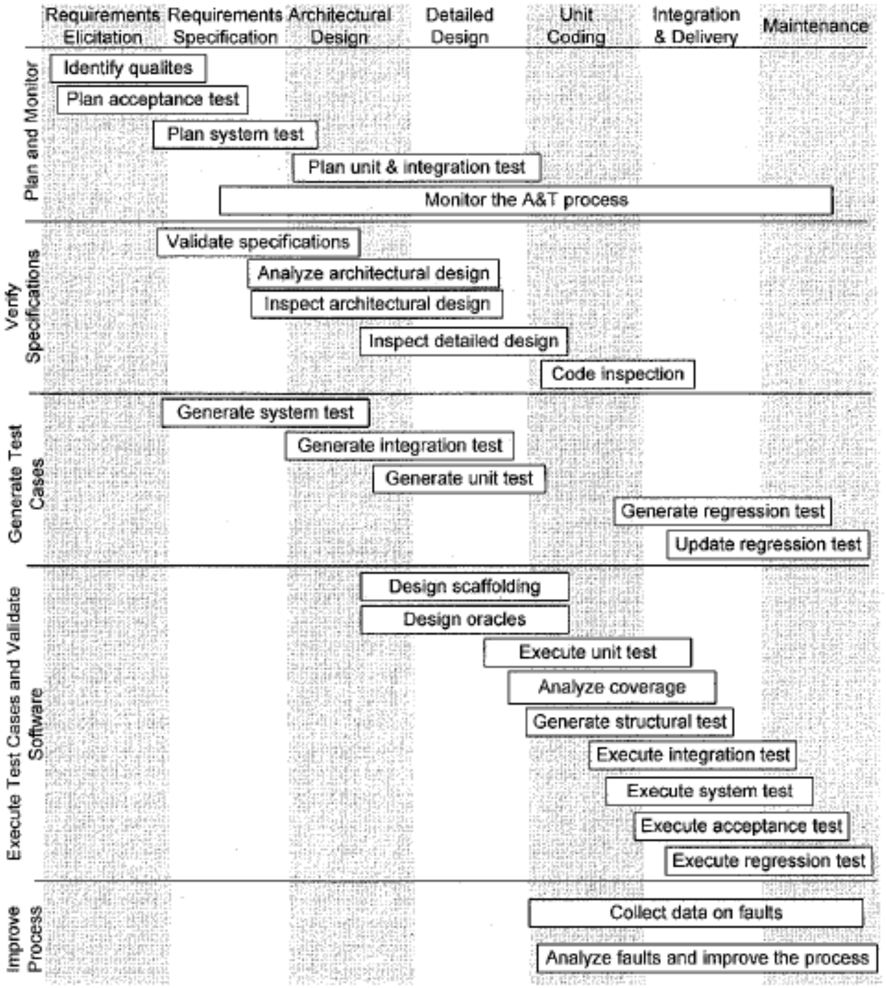


Figure : Main analysis and testing activities through the software life cycle. Source: Pezzè and Young (reference book)

What techniques should be applied during software development?

- Feasibility study to understand and select appropriate techniques
- Verification to lead development according to requirements
- Validation to check product against users' expectations

What techniques should be applied during software development?

- Techniques depend on quality, cost, scheduling, resources …
  - No "one technique"
  - Effectiveness for different classes of faults to capture
  - Applicability at different stages of the project (early stage artefacts cannot be tested with automated tools)
  - Techniques can have different goals (to understand the coverage or to detect faults)
  - Trade-off cost and assurance, e.g, focus on key few properties

What techniques should be applied during software development?

- Feasibility study includes
  - Tentative architectural design to modularise the work and identify properties that can be verified in different subsystems
  - Draft plan to break projects into incremental deliveries preliminary decisions about test and analysis techniques

How can we assess the readiness of a product to be released?

- For example, with measurement:
  - **Availability**: e.g., daily down time
  - **Time Between Failures**: days between two consecutive failures
  - **Defect slippage**: number of defects remaining in the product

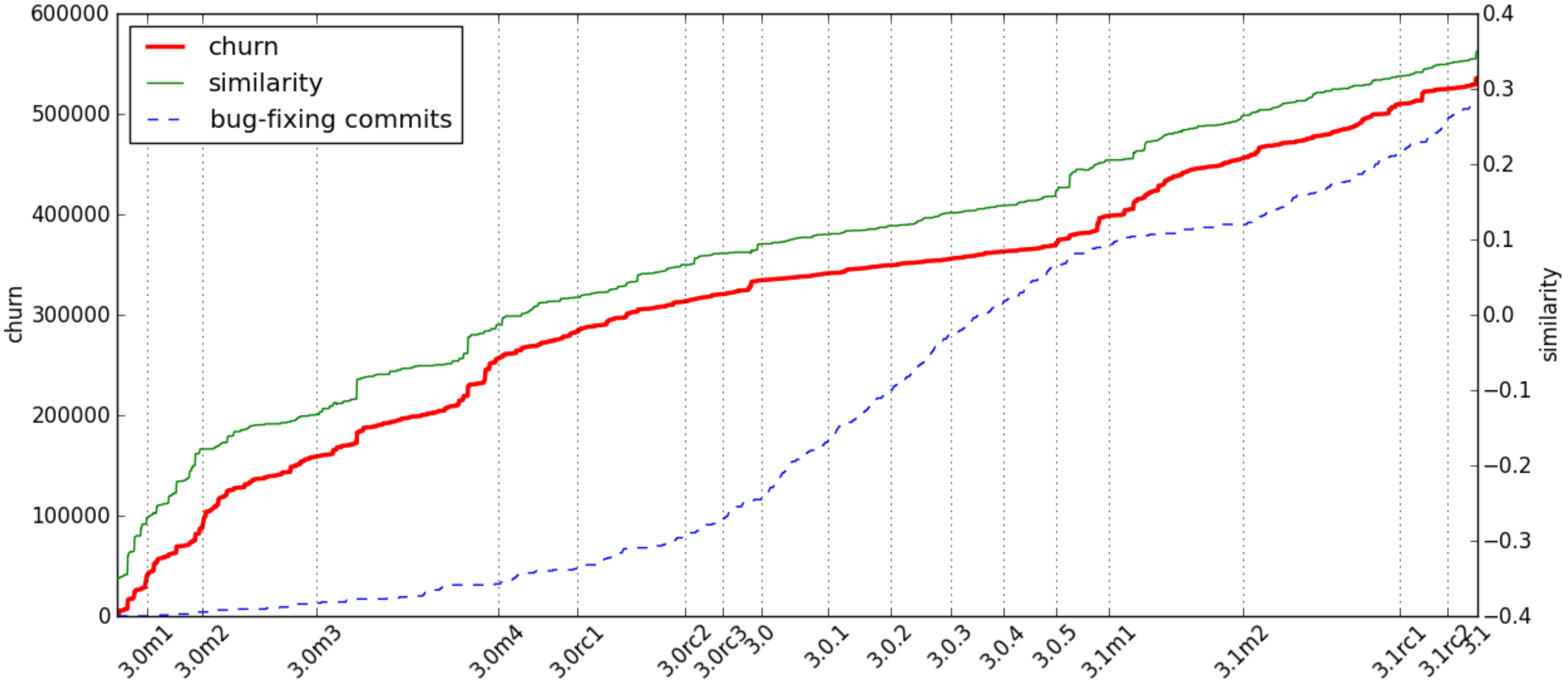
How can we assess the readiness of a product to be released?

- How:
  - **alpha testing**: performed by users in a controlled environment. Capture operational profiles decide by the organisation
  - **beta testing**: performed by users in a their own environment. Capture different operational profiles

How can we control the quality of successive releases?

- Major revisions also called “point releases”: the full quality process is repeated including beta testing and regression testing
- Small revisions also called “patch releases”: revisions to incorporate bug fixing. Characterised by fast/automated testing. Subset of regression testing.
- In Open Source Software we might have “milestones,” “release candidates,” and “major releases”

How can we control the quality of successive releases?



How can the development process be improved?

- Data collection across projects for modelling and prediction, e.g., faults and their severity or priority
- Team/Personal Software Process (Humphrey)