LECTURE 09.
BUG REPORTING

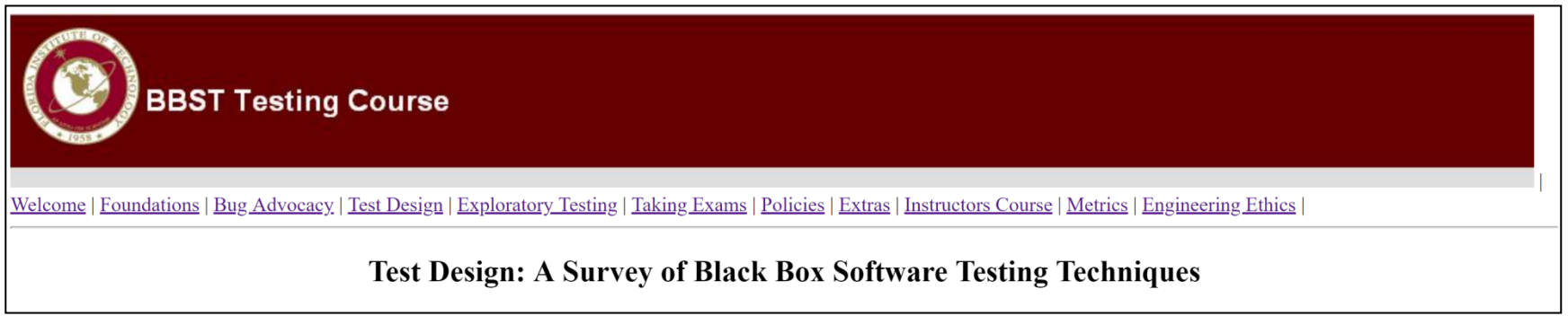
Test Design Techniques
[20 April 2022]

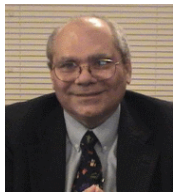
Elective Course, Spring Semester, 2021-2022

Camelia Chisăliţă-Creţu, Lecturer PhD
Babeş-Bolyai University

Acknowledgements

The course Test Design Techniques is based on the Test Design course available on the **BBST Testing Course** platform.





The BBST Courses are created and developed by **Cem Kaner, J.D., Ph.D.., Professor of Software Engineering at Florida Institute of Technology.**

Contents

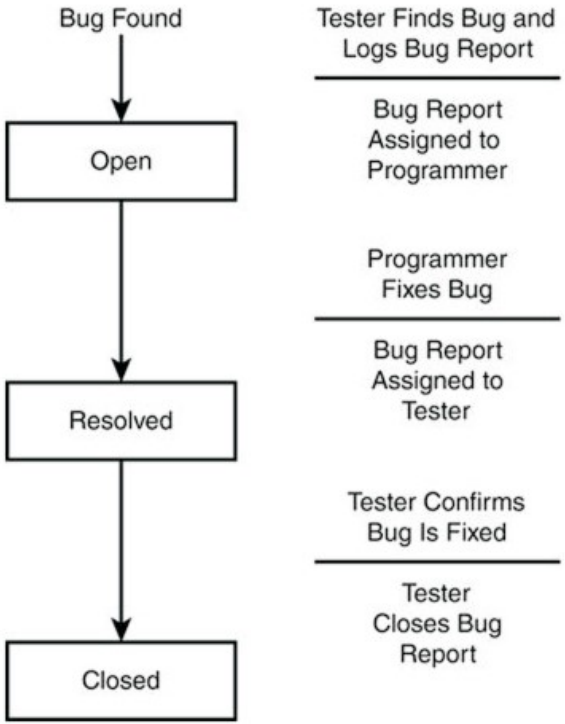
- Last lecture…
  - **Activity-based techniques**
- Terminology
- **Bug Reporting**
  - RIMGEA;
  - Type of Bugs
    - Coding Bugs;
    - Design Bugs;
  - Examples;
  - Quality-based Bug Taxonomy.

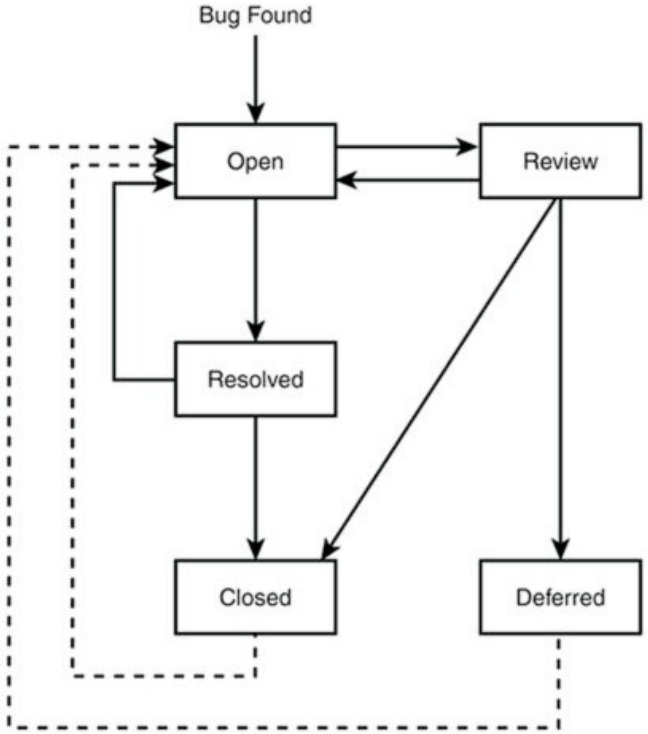
BUG LIFE CYCLE

Bug Life Cycle Approaches

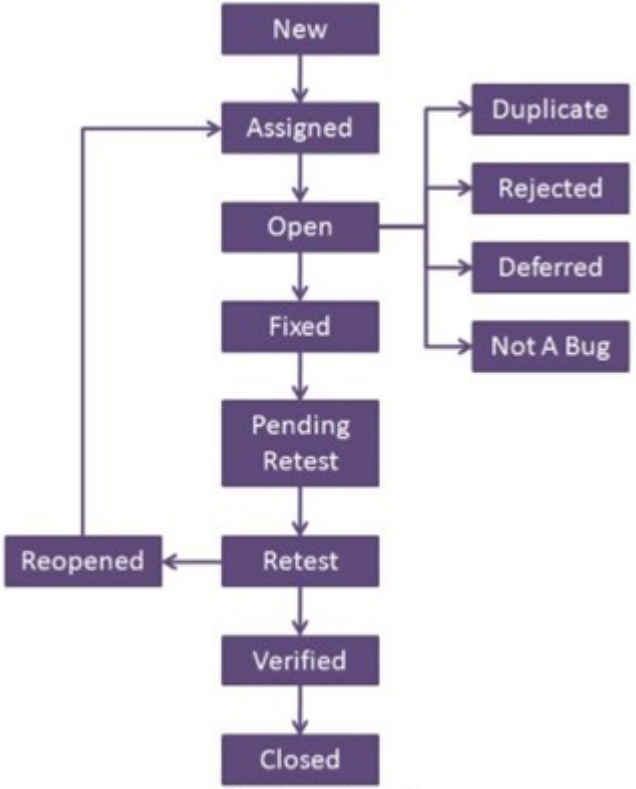
Good practices for bug reporting

Bug Life Cycle (1)





Bug Life Cycle (2)



Bug / Defect Lifecycle

- Bug states that are set by **tester**:
  - **New, Pending Testing, Retest, Reopened, Verified, Closed;**
- Bug states that are set by the **developer**:
  - **Assigned, Open (Duplicate, Rejected, Deffered, Not a Bug), Fixed;**

[ISTQBCertification2022]

Good Practices for Bug Reporting (1)

- **Principles for bug reporting [Patton2005]:**
  - the bug should be reported immediately after it was identified;
  - the bug description should be provided;
  - no personal remarks related to thhe bug should be made by the bug reporter;
  - the bug state is monitored by the tester after it was reported, i.e., a bug tracking systems is used.

RIMGEA

Definition. Components. Applicability. Goals. Conceptual Representation

Replicate. Isolate. Maximize. Generalize. Externalize. Communicate clearly

Coding/Implementation Bug vs. Design Bug

Examples

RIMGEA. Definition. Components

- **RIMGEA** [BBST2008]
  - group of rules applied in order to investigate and improve a bug description;

- **components**
  - **R**eplicate – the issue/bug/error is reproduced;
  - **I**solate – the bug is isolated;
  - **M**aximize – the bug effect is maximized;
  - **G**eneralize – the bug is generalized;
  - **E**xternalize – the bug is externalized;
  - **A**nd say it clear and dispassionately – use a neutral tone when the bug is reported.

RIMGEA. Applicability

- RIMGEA rules are applied in order to report:
  - **implementation/coding bugs**
    - **the software behaves in a way the designer or programmer will agree is improper, inadequate;**
    - it exposes a defect, i.e., mistake in coding the adopted solution;
  - **design bugs**
    - **the software behaves in a way the designer and the programmer intended, still it may indicate a solution that may be improved;**
    - it does not expose an implementation defect, but a flaw approach to provide the adopted solution.

RIMGEA. Goals

- RIMGEA is used for:
  - **reporting a coding bug**
    - to build a report that consists of a **minimal step list that certifies the bug existence** within the source code;
  - **reporting a design bug**
    - to build a report that **clearly states the design aspect that represents an issue** and **describes the way it reduces the software quality.**

RIMGEA. Conceptual representation

Replicate

Isolate

Maximize

Generalize

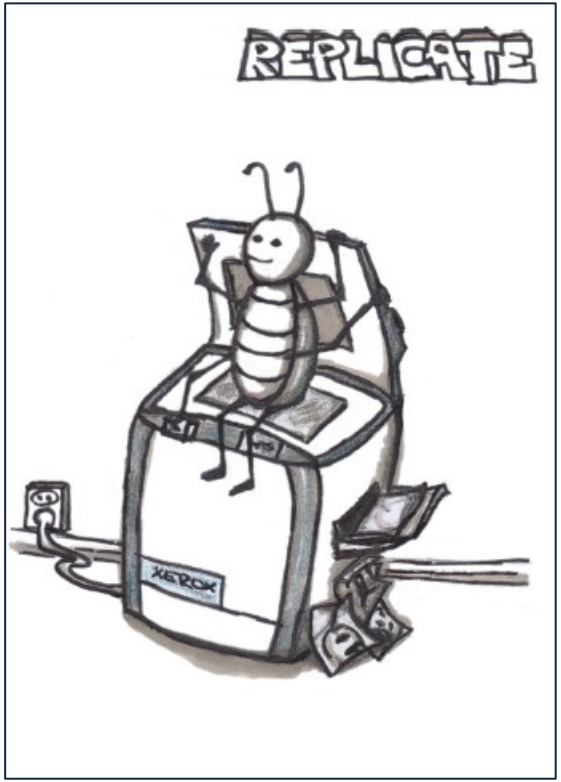
Externalize

Communicate clearly

sursa: [Altom2016]

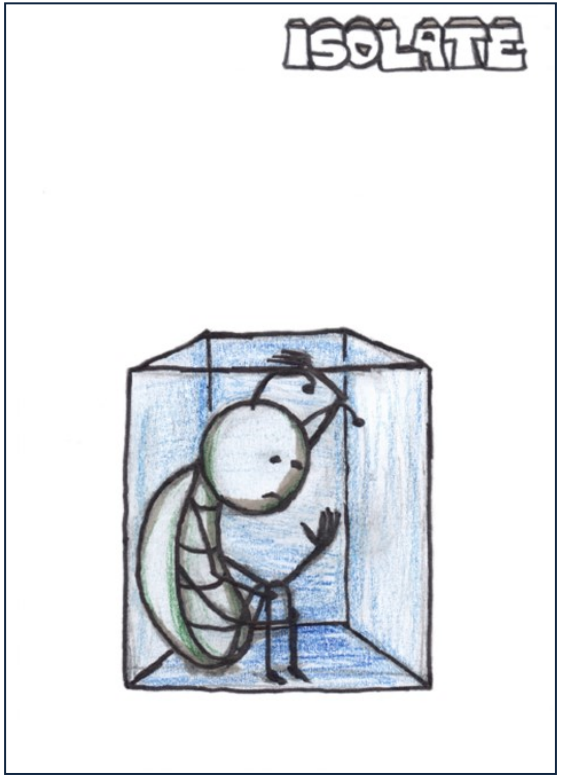
RIMGEA. **R**eplicate

- **replicate** (*rom.* **reproducerea bug-ului**):
  - testing activity that states **what** is required in order to reproduce the bug each time someone intends to show its presence and manifest its consequences;
  - it emphasizes cases when the bug **cannot** be reproduces and describes the factors that may increase the probability of its appearance.



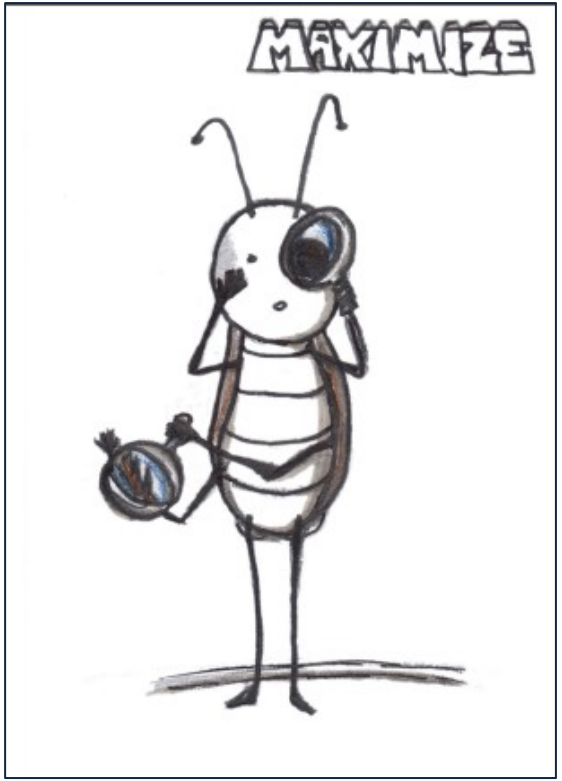
RIMGEA. Isolate

- **isolate** (*rom.* **izolarea bug-ului**):
  - testing activity that identifies **the shortest sequence of steps necessary to reproduce the bug** and to report it in a clear manner, such that the failure is exposed;
  - a bug report consists of the description of a single failure within the software product.



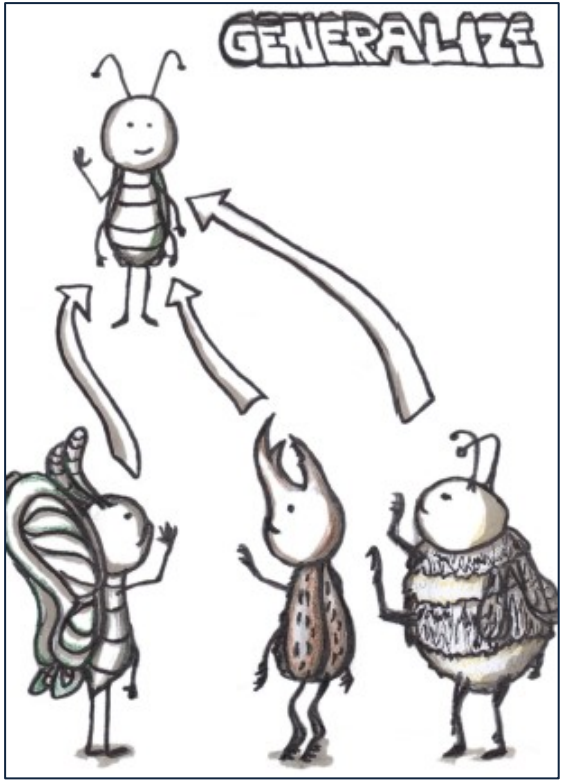
RI**M**GEA. **M**aximize

- **maximize** (*rom.* **maximizarea efectului bug-ului**):
  - testing activity that identifies **the most important consequences of the bug's existence**.



RIM**G**EA. **G**eneralize

- **generalize** (*rom.* **generalizarea efectului bug-ului**):
  - testing activity that determine **the range of circumstances under which this bug will cause a failure**;
  - it is recommended to show the **problems caused by this bug will show up often**, on many common configurations, when processing realistic data.



RIMG**E**A. **E**xternalize

- **externalize** (*rom.* **externalizarea efectului bug-ului**):
  - testing activity that considers the consequences of the bug, following various perspectives:
    - **actual user:** How it will affect the actual user?
    - **client/stakeholder:** How it will affect the client's reputation and/or business?
    - **provider/developer:** How it will affect the provider's reputation?
    - **competition/third party:** How important is the bug?

RIMGE**A**. **A**nd use a constructive communication

- **And say it clearly and dispassionately** (*rom.* **comunicare clară şi constructivă/neutră**):
  - *the bug report does consists of* an **unbiased approach** when addressing the factual or speculative issues;
  - *the bug report does not include*
    - **things that may irritate or criticize people;**
    - **irrelevant aspects** that may affect other people, unless they are essential to a proper problem description and the circumstances that cause the failure.

Coding Bug vs. Design Bug

RIMGEA	Coding Bug	Design Bug
Replicate	Essential	Rarely important
Isolate	Important	Important
Maximize	Sometimes essential	Useful
Generalize	Important	Useful
Externalize	Sometimes useful	Essential
And say it in a constructive way	Essential	Essential

RIMGEA. Examples

- SUT:
  - Apache Open Office: https://www.openoffice.org/;
- bug management tool
  - Bugzilla: https://bz.apache.org/ooo/;

- running configuration used for testing (that allowed to identify the bugs):
  - **Operating system:** Windows 10 Home, 64-bit;
  - **processor:** Intel Core i5, 1.7GHz;
  - **RAM:** 4Gb;
  - **Open Office Writer version:** 4.1.3.

Open Office Writer. Investigated Issues

- **coding/implementation bugs:**
  - **Bug 1. rounding error:**
    - **Issue 120368 - Font size with decimal values don't have a consistent approximation** (https://bz.apache.org/ooo/show_bug.cgi?id=120368);
  - **Bug 2. setting inconsistency:**
    - **Issue 127562 - Inconsistency on Header Height max. value when enabled first time** (https://bz.apache.org/ooo/show_bug.cgi?id=127562);
- **design bugs:**
  - **Bug 3. design inconsistency:**
    - **Issue 126371 – Disappearing Vertical Text button in Drawing Toolbar in Writer v.4.1.1** (https://bz.apache.org/ooo/show_bug.cgi?id=126371).

Open Office Writer. Bug 1 (1)

- **Bug 1. rounding error:**
  - **Issue 120368 - Font size with decimal values don't have a consistent approximation** (https://bz.apache.org/ooo/show_bug.cgi?id=120368);

  - **R**eplicate – the bug should be able to be reproduced;
  - **I**solate – the bug should be isolated in a smallest sequence of steps;
  - **M**aximize – the bug consequences should be maximized;
  - **G**eneralize – the bug range of circumstances should be generalized;
  - **E**xternalize – the bug impact on stakeholder, company reputation and third party is addressed;
  - **A**nd say it clear and dispassionately – use a constructive approach and a neutral tone when reporting the bug.

Open Office Writer. Bug 1 (2)

- **Bug 1. rounding error:**
  - **Issue 120368 - Font size with decimal values don't have a consistent approximation** (https://bz.apache.org/ooo/show_bug.cgi?id=120368);

- **R**eplicate – the bug should be able to be reproduced;
  - **Scenario A:**
    - Current Size: 20;
    - Main window of AOO Write, setting the Size to 10.23:
      - Main window: Size = 10.1;
      - Format window: Size = 10.2;
  - **Scenario B:**
    - Current Size: 20;
    - Window Format, setting the Size to 10.23:
      - Format window: Size = 10.2;
      - Main window: Size = 10.1;

Open Office Writer. Bug 1 (3)

- **Bug 1. rounding error:**
  - **Issue 120368 - Font size with decimal values don't have a consistent approximation** (https://bz.apache.org/ooo/show_bug.cgi?id=120368);

- **I**solate – the bug should be isolated in a smallest sequence of steps;
  - the smallest sequence of steps that reproduce the bug should be provided;
    - E.g., Scenario A, Scenario B.

Open Office Writer. Bug 1 (4)

- **Bug 1. rounding error:**
  - **Issue 120368 - Font size with decimal values don't have a consistent approximation** (https://bz.apache.org/ooo/show_bug.cgi?id=120368);

- **M**aximize – the bug consequences should be maximized;
  - Maximize the negative impact:
    - **negative impact = inconsistency when rounding the Font Size when using the two scenarios;**
    - the rounding error itself is less important compared to the inconsistency of handling the rounding operation in different scenarios, e.g., Scenario A, Scenario B:
      - **the Font Size attribute seems to have assigned to different values 10.1 and 10.2; which one is the correct one?**

Open Office Writer. Bug 1 (5)

- **Bug 1. rounding error:**
  - **Issue 120368 - Font size with decimal values don't have a consistent approximation** (https://bz.apache.org/ooo/show_bug.cgi?id=120368);

- **G**eneralize – the bug range of circumstances should be generalized;
  - **all platforms allow to reproduce this bug:**
    - the rounding error approximation is very small;
    - **only a few users make actually use of the Font Size attribute with decimals.**

Open Office Writer. Bug 1 (6)

- **Bug 1. rounding error:**
  - **Issue 120368 - Font size with decimal values don't have a consistent approximation** (https://bz.apache.org/ooo/show_bug.cgi?id=120368);

•**E**xternalize – the bug impact on stakeholder, company reputation and third party is addressed;

- when the actual user identifies the rounding error he becomes suspicious about the correctness of other approximation roundings computations available in AOO Writer;
- if Scenario B provides the correct approximation from the user's perspective, then his user experience is hindered as he needs to remind himself that each time he sets the Font Size attribute, Scenario A is not appropriate and he should be avoided;
- **the tester needs to check how the competition, i.e., Microsoft Office Word, addresses the issue of rounding approximation in general and Font Size attribute in particular.**

Open Office Writer. Bug 2 (1)

- **Bug 2. setting inconsistency:**
  - **Issue 127562 - Inconsistency on Header Height max. value when enabled first time** (https://bz.apache.org/ooo/show_bug.cgi?id=127562);

  - **R**eplicate – the bug should be able to be reproduced;
  - **I**solate – the bug should be isolated in a smallest sequence of steps;
  - **M**aximize – the bug consequences should be maximized;
  - **G**eneralize – the bug range of circumstances should be generalized;
  - **E**xternalize – the bug impact on stakeholder, company reputation and third party is addressed;
  - **A**nd say it clear and dispassionately – use a constructive approach and a neutral tone when reporting the bug.

Open Office Writer. Bug 2 (2)

- **Bug 2. setting inconsistency:**
  - **Issue 127562 - Inconsistency on Header Height max. value when enabled first time**

  (https://bz.apache.org/ooo/show_bug.cgi?id=127562);

- **R**eplicate – the bug should be able to be reproduced;

Scenario A

- 1. Open OpenOffice Writer;
- 2. Set the measurement unit to 'cm': Tools menu | Options...| OpenOffice Writer | General | Choose as Measurement Unit 'Centimeters';
- 3. Create a new document: Menu File | New | Text Document;
- 4. Set the Paper Format to A4: Format Menu | Page...| Page tab | Select from Format drop-down list;
- 5. go to Header Tab;
- 6. Enable Header;
- 7. Set Header Height to 29.7 cm;
- 8. Change focus from Hearder Height; [header height = 20.06 cm];
- 9. Repeat steps 6 and 7; [header height = 20.56 cm].

Open Office Writer. Bug 2 (3)

- **Bug 2. setting inconsistency:**
  - **Issue 127562 - Inconsistency on Header Height max. value when enabled first time**

  (https://bz.apache.org/ooo/show_bug.cgi?id=127562);

- **R**eplicate – the bug should be able to be reproduced;

Scenario B

- 1. Open OpenOffice Writer;
- 2. Set the measurement unit to 'cm';
- 3. Create a new document;
- 4. keep the default Paper Format to Letter;
- 5. go to Header Tab;
- 6. Enable Header;
- 7. Set Header Height to 27.94 cm;
- 8. Change focus from Header Height; [header height = 18.65 cm]
- 9. Repeat steps 7 and 8; [header height = 19.15 cm]
- 10. go to Page tab again;
- 11. Set the Paper Format to A4;
- 12. go to Header Tab;
- 13. Set Header's Height to 29.7 cm;
- 14. Change focus from Header Height; [header height = 20.56 cm];
- 15. Repeat steps 12 and 13; [header height = 20.56 cm].

Open Office Writer. Bug 2 (4)

- **Bug 2. setting inconsistency:**
  - **Issue 127562 - Inconsistency on Header Height max. value when enabled first time** (https://bz.apache.org/ooo/show_bug.cgi?id=127562);
- **I**solate – the bug should be isolated in a smallest sequence of steps;
  - **Scenario A:**
    - it provides the smallest sequence of steps the reproduces the issue;
  - **Scenario B:**
    - it indicates the bug shows up regardless of the initial page format, i.e., A4 in scenario A, Letter and then A4 in scenario B;
    - **it suggests the issue shows up on the first activation of the Header Height option in the Page Style window settings.**

Open Office Writer. Bug 3 (1)

- **Bug 3. design inconsistency:**
  - **Issue 126371 – Disappearing Vertical Text button in Drawing Toolbar in Writer v.4.1.1**

  (https://bz.apache.org/ooo/show_bug.cgi?id=126371);

Design issue investigation focus on the appropriateness of the behaviour not if the behaviour is correct or not.

- **R**eplicate – the bug should be able to be reproduced;
- **I**solate – the bug should be isolated in a smallest sequence of steps;
- **M**aximize – the bug consequences should be maximized;
- **G**eneralize – the bug range of circumstances should be generalized;
- **E**xternalize – the bug impact on stakeholder, company reputation and third party is addressed;
- **A**nd say it clear and dispassionately – use a constructive approach and a neutral tone when reporting the bug.

Open Office Writer. Bug 3 (2)

- **Bug 3. design inconsistency:**
  - **Issue 126371 – Disappearing Vertical Text button in Drawing Toolbar in Writer v.4.1.1**

  (https://bz.apache.org/ooo/show_bug.cgi?id=126371);

- **R**eplicate – the bug should be able to be reproduced;
  - button Vertical Text:
    - it is available when Asian Language Support from Tools Options | Language Settings | Languages is activated , which allows to write top to bottom, vertically.

Open Office Writer. Bug 3 (3)

- **Bug 3. design inconsistency:**
  - **Issue 126371 – Disappearing Vertical Text button in Drawing Toolbar in Writer v.4.1.1**

  (https://bz.apache.org/ooo/show_bug.cgi?id=126371);

- **I**solate – the bug should be isolated in a smallest sequence of steps;
  - **Scenario A**
    - Asian Language Support **is not activated**;
    - the user tries to add button Vertical Text from Drawing toolbar, by checking it in the Customize window; **effect:** the button appears and disappears right away;
    - if the button Vertical Text from the Drawing toolbar is already checked in the Customize window, the button does not appear in the Drawing toolbar, as the language setting prevails.

Open Office Writer. Bug 3 (4)

- **Bug 3. design inconsistency:**
  - **Issue 126371 – Disappearing Vertical Text button in Drawing Toolbar in Writer v.4.1.1**

  (https://bz.apache.org/ooo/show_bug.cgi?id=126371);

- **I**solate – the bug should be isolated in a smallest sequence of steps;
  - **Scenario B**
    - Asian Language Support **is activated**;
    - the user tries to remove the button Vertical Text from Drawing toolbar, by un-checking it in the Customize window; **effect:** the button disappears and appears right away;
    - if the button Vertical Text from the Drawing toolbar is not checked in the Customize window, it still appears in the Drawing toolbar, as the language settings prevails.

Open Office Writer. Bug 3 (5)

- **Bug 3. design inconsistency:**
  - **Issue 126371 – Disappearing Vertical Text button in Drawing Toolbar in Writer v.4.1.1**

  (https://bz.apache.org/ooo/show_bug.cgi?id=126371);

- **E**xternalize – the bug impact on stakeholder, company reputation and third party is addressed;
  - impact on the actual user:
    - the product quality is affected, as the user becomes frustrated when failing to add/remove the button vertical text from the Drawing toolbar;
    - competition and/or third party products, e.g., Microsoft Office Word, do not constrain to use the button Vertical Text when specific language settings are available only;
      - the product quality is affected by forcing the user to remind himself to change the language settings first in order to achieve vertical text writing and, after performing the intended actions to go back to initial language settings;
      - if the user forgets to go back to the initial settings, he might find difficult to use the configurations that are implied by the language settings ==> the user does not benefit from product features in the desired manner.

QUALITY-BASED BUG TAXONOMY

by Claudiu Draghia

Quality Attributes and their Bugs (1)

- **Functionality** - **Functionality bug**;
  - *Almost human like shape;*
  - *Curious with funny eyes;*
  - *Dressed to impress.*

source [Draghia2019]

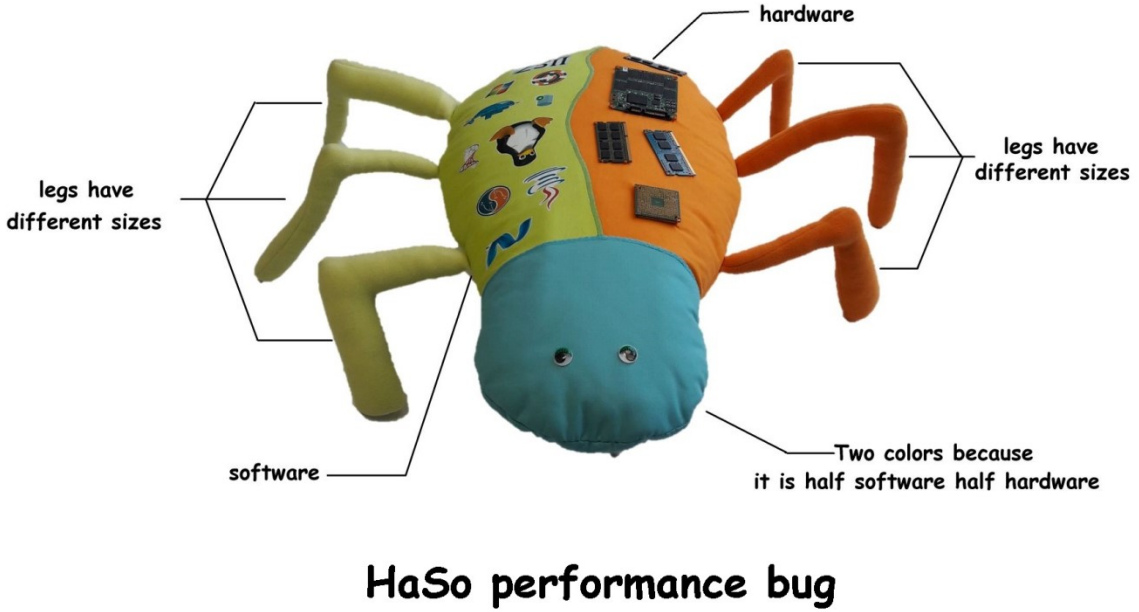
Quality Attributes and their Bugs (2)

- **Testability** - **CHAOS UI**;
  - *Confident eyes;*
  - *Camouflaged;*
- **C**ontrollability;
- **H**eterogeneity;
- **A**utomatability;
- **O**bservability;
- **S**eparation of concerns;
- **U**nderstandability;
- **I**solateability.

source [Draghia2019]

Quality Attributes and their Bugs (3)

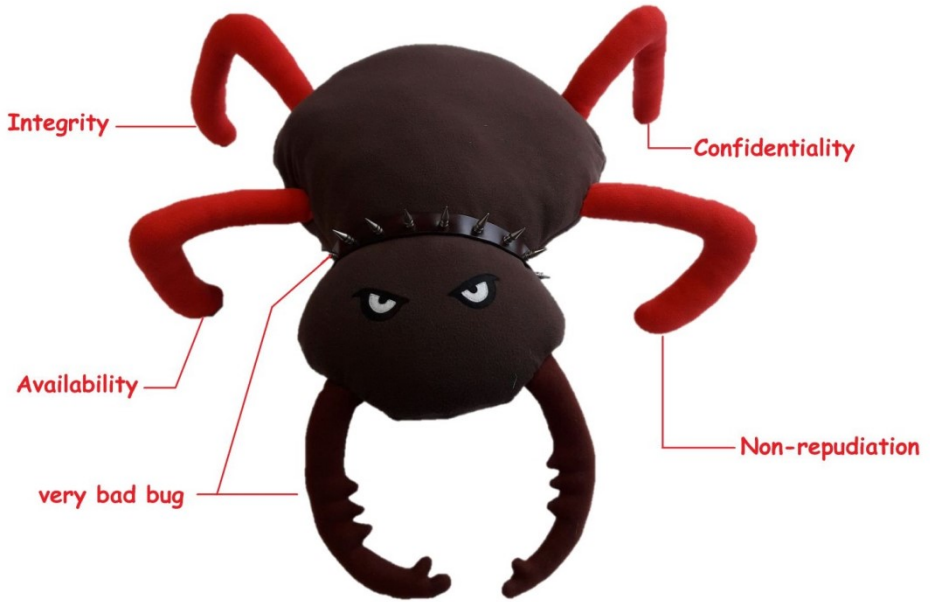
- **Performance** - **HaSo**;
  - *Two part bug:*
    - **Ha**rdware;
    - **So**ftware;



source [Draghia2019]

Quality Attributes and their Bugs (4)

- **Security** – **CIAn**;
  - *very bad bug;*
- **C**onfidentiality;
- **I**ntegrity;
- **A**vailability;
- **n**on-repudiation.



source [Draghia2019]

Lecture Summary

- We have discussed:
  - Activity-based techniques;
  - Bug reporting;
    - RIMGEA approach;
    - Coding Bugs and Design Bugs:
      - Examples;
  - Quality-based Bug Taxonomy.

Lecture 10

- **Evaluation-based Techniques:**
  - Function equivalence testing;
  - Mathematical oracle;
  - Constraint checks;
  - Self-verifying data;
  - Comparison with saved results;
  - Comparison with specifications or other authoritative documents;
  - Diagnostics-based testing;
  - Verifiable state models.
- **Desired-results Techniques:**
  - Build verification;
  - Confirmation testing;
  - User acceptance testing;
  - Certification testing.

References

- **[BBST2011]** BBST – Test Design, Cem Kaner, http://www.testingeducation.org/BBST/testdesign/BBSTTestDesign2011pfinal.pdf.
- **[BBST2010]** BBST – Fundamentals of Testing, Cem Kaner, http://www.testingeducation.org/BBST/foundations/BBSTFoundationsNov2010.pdf.
- **[ISTQBCertification2022]** ISTQB Exam Certification, http://istqbexamcertification.com/what-is-a-defect-life-cycle/.
- **[Patton2005]** R. Patton, *Software Testing,* Sams Publishing, 2005.
- **[BBST2008]** Black-Box Software Testing (BBST), Bug Advocacy, http://www.testingeducation.org/BBST/bugadvocacy/BugAdvocacy2008.pdf.
- **[Altom2016]** Levente Balint, *BLOG : RIMGEN, How Well Do you Advocate For Your Bugs?*, http://altom.training/blog/tag/rimgen/.
- **[Draghia2019]** Claudiu Draghia, http://bugs.brainforit.com/.