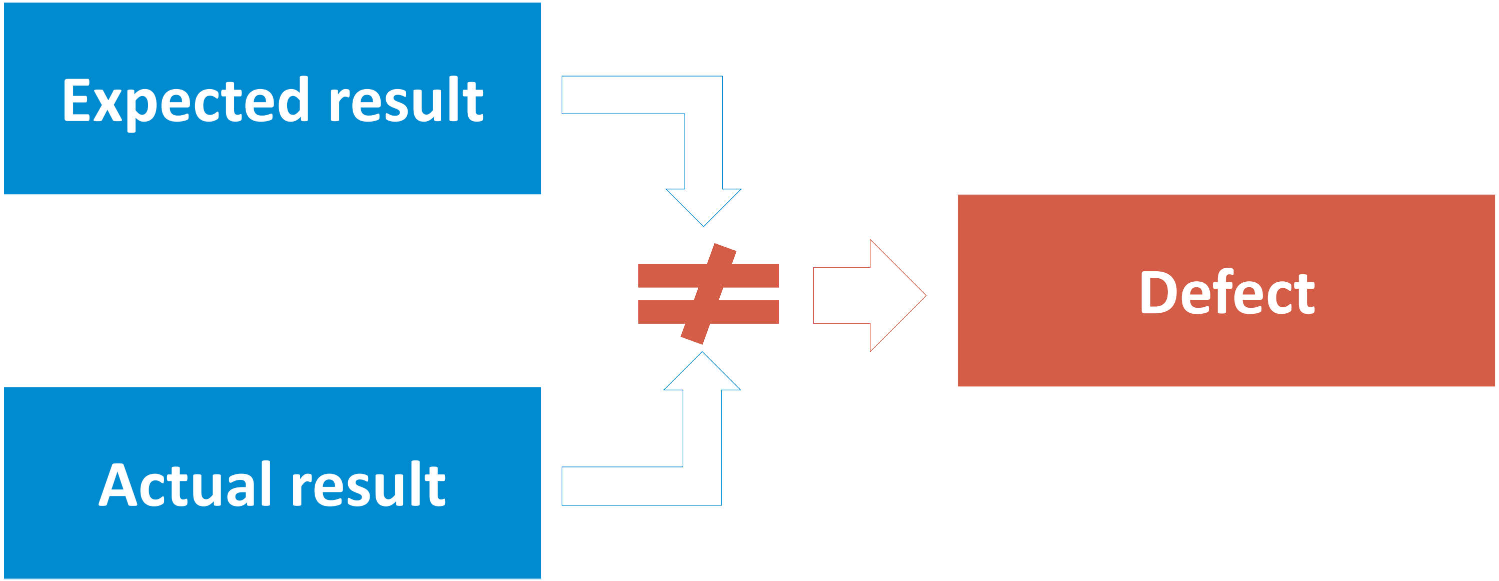
Defects and Defect Reports

Software Testing Introduction

TRAINING

CENTER

Definitions



Definitions

Read and remember!

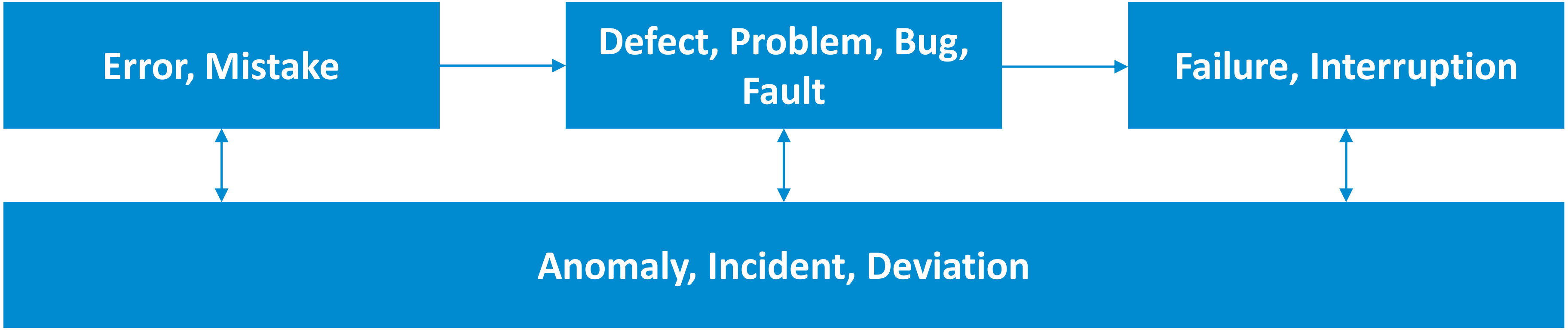
Defect – an imperfection or deficiency in a work product where it does not meet its requirements or specifications.

Expected result – the predicted observable behavior of a component or system executing under specified conditions, based on its specification or another source.

Actual result – the behavior produced/observed when a component or system is tested.

More ~~scientific~~ technical approach

Read and remember!



Anomaly, incident, deviation – any deviation of the observed (actual) result, state, behavior, value, property from the observer's expectations, formed on the basis of requirements, specifications, other documentation or experience and common sense.

Defect Report

Read and remember!

Defect report – documentation of the occurrence, nature, and status of a defect.

It’s important!

A defect report should…

Describe the issue **in details**

Define the **severity and priority** of the issue

Provide **all necessary data** to reproduce and fix the issue

Help developer to fix the issue

It’s important!

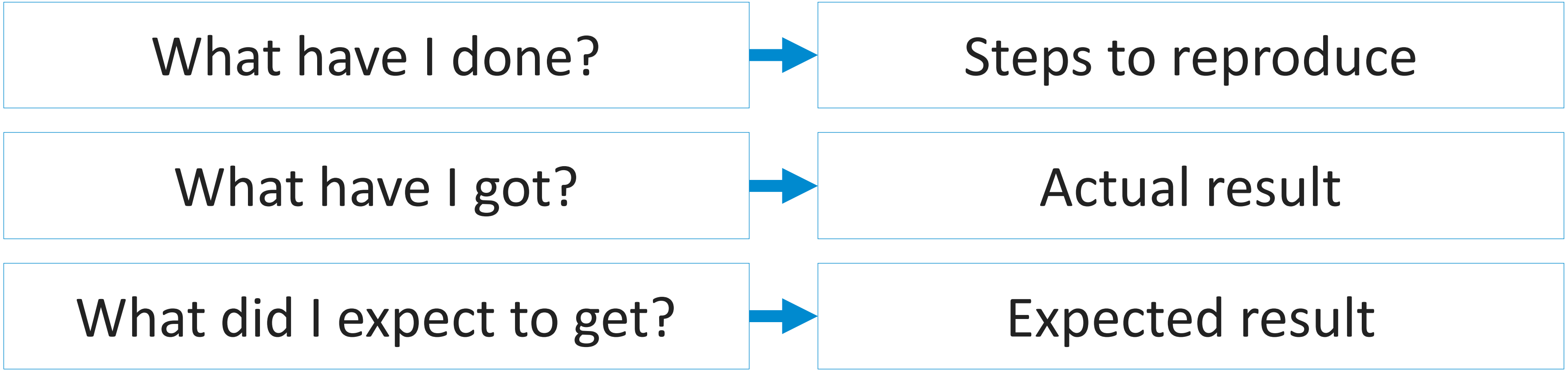
Bad defect report	**Good defect report**
Lacks significant details	Provides significant details
Takes time to understand	Easy and obvious
Forces the developer to re-do testers work	Facilitates quick and easy solutions

It’s important!

The main goal of a defect report – is to get the defect fixed

Some simple logic

Before you write down a defect report, make sure you have answers to the following questions

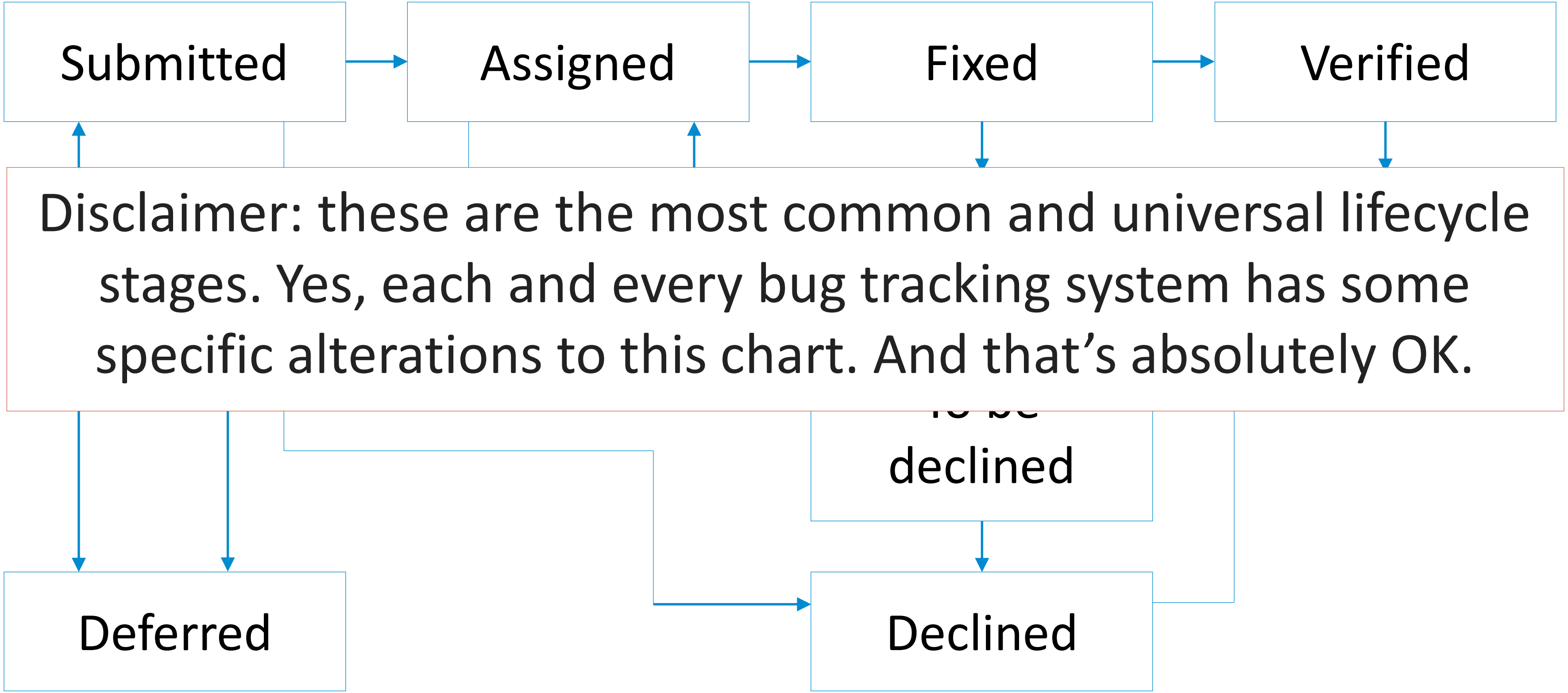


What have I done? → Steps to reproduce

What have I got? → Actual result

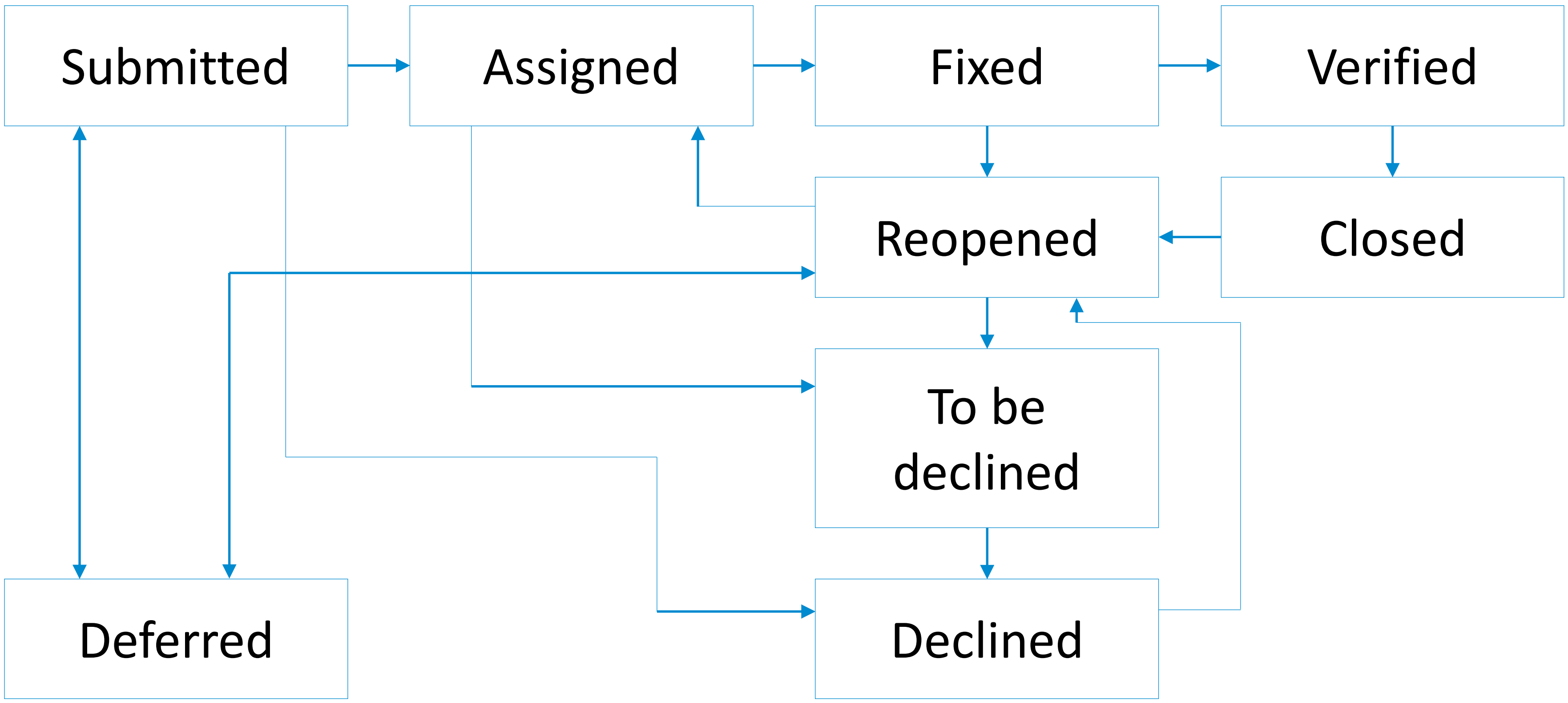
What did I expect to get? → Expected result

Defect Report Lifecycle



Disclaimer: these are the most common and universal lifecycle stages. Yes, each and every bug tracking system has some specific alterations to this chart. And that’s absolutely OK.

Defect Report Lifecycle



Submitted

Assigned

Fixed

Verified

Reopened

Closed

To be declined

Deferred

Declined

Defects and Defect Reports

Software Testing Introduction

TRAINING CENTER

Defect Report Fields – Part 1

Software Testing Introduction

TRAINING CENTER

Key defect report fields

Disclaimer: each and every bug tracking system has much more fields to fill. There are different approaches in details, but now we are going to review only **KEY FIELDS** that remains more or less the same for past 3-4 decades.

Key defect report fields: general overview

ID	Summary	Description	Steps to reproduce
19	Infinite loop on input file with read-only attribute	If an input file has the “read-only” attribute, the app can not move the processed file into the destination directory: so the app processes the file again and again and thus falls into the infinite loop. **Exp**: the processed file is moved from the input directory to the destination directory. **Act:** the processed data (new file) appears inside the destination directory, but the original file is not deleted from the input directory. **Req:** DS-2.1.	1. Place a valid file (size, type) into the input directory. 2. Set the “read-only” attribute on this file. 3. Start the app. **Bug:** the processing result appears inside the destination dir (and the file is repeatedly updated according to the last write time), but the original file stays inside the input directory.

Reproducibility	Severity	Priority	Symptom	Workaround	Comments	Attachments
Always	Medium	Normal	Incorrect operation	No	If the customer has no special plans for using “read-only” attribute on files in the input directory, the Easiest solution is to remove the attribute once it is detected.	-

Key defect report fields: ID

ID

19	Infinite loop on input file with read-only attribute	If an input file has the “read-only” attribute, the app can not move the processed file into the destination directory: so the app processes the file again and again and thus falls into the destination directory, but the original file is not deleted from the input directory. **Req:** DS-2.1.	1. Place a valid file (size, type) into the input directory. 2. Set the “read-only” attribute on this file. 3. Start the app. **Bug:** the processing result appears inside the destination dir (and the file is repeatedly updated according to the last write time), but the original file stays inside the input directory.

- Unique.
- Meaningful (if bug tracking system allows).

Always	Medium	Normal	Incorrect operation	No	If the customer has no special plans for using “read-only” attribute on files in the input directory, the Easiest solution is to remove the attribute once it is detected.	-

Key defect report fields: summary

Summary

19	Infinite loop on input file with read-only attribute	If an input file has the “read-only” attribute, the app can not move the processed file into the destination directory: so the app processes the file again and again and thus falls into the infinite loop. **Exp**: the processed file is moved from the input directory to	1. Place a valid file (size, type) into the input directory. 2. Set the “read-only” attribute on this file. 3. Start the app.

- Answers questions: **<u>what</u>** did happen, **<u>where</u>** did it happen, and in **<u>what conditions</u>** did it happen.
- Should at the same time:
  - Provide as much information as possible.
  - Be as short as possible.
  - Be easily distinguishable from other summaries.

Al

remove the attribute once it is detected.

Key defect report fields: description

Description

19	Infinite loop on input file with read-only attribute	If an input file has the “read-only” attribute, the app can not move the processed file into the destination directory: so the app processes the file again and again and thus falls into the infinite loop. **Exp**: the processed file is moved from the input directory to the destination directory. **Act:** the processed data (new file) appears inside the destination directory, but the original file is not deleted from the input directory. **Req:** DS-2.1.	1. Place a valid file (size, type) into the input directory. 2. Set the “read-only” attribute on this file. 3. Start the app. **Bug:** the processing result appears inside the destination dir (and the file is repeatedly updated according to the last write time), but the original file stays inside the input directory.

Al

- Contains detailed defect description.
- Unlike Summary, Description may be long enough.
- In many BTS testers use Description to write down the actual result, the expected result and the reference to the corresponding requirement.

Key defect report fields: steps to reproduce

Steps to reproduce

19	Infinite loop on input file with read-only attribute	If an input file has the "read-only" attribute, the app can not move the processed file into the destination directory: so the app processes the file again and again and thus falls into the infinite loop. **Exp**: the processed file is moved from the input directory to the destination directory. **Act:** the processed data (new file) appears inside the destination directory, but the original file is not deleted from the input directory. **Req:** DS-2.1.	1. Place a valid file (size, type) into the input directory. 2. Set the "read-only" attribute on this file. 3. Start the app. **Bug:** the processing result appears inside the destination dir (and the file is repeatedly updated according to the last write time), but the original file stays inside the input directory.

- Contains detailed description of actions to be done to reproduce the defect.
- May contain a short description of the defect or the final erroneous state of the application.

Al

Key defect report fields: general overview

ID	Summary	Description	Steps to reproduce
19	Infinite loop on input file with read-only attribute	If an input file has the “read-only” attribute, the app can not move the processed file into the destination directory: so the app processes the file again and again and thus falls into the infinite loop. **Exp**: the processed file is moved from the input directory to the destination directory. **Act:** the processed data (new file) appears inside the destination directory, but the original file is not deleted from the input directory. **Req:** DS-2.1.	1. Place a valid file (size, type) into the input directory. 2. Set the “read-only” attribute on this file. 3. Start the app. **Bug:** the processing result appears inside the destination dir (and the file is repeatedly updated according to the last write time), but the original file stays inside the input directory.

Reproducibility	Severity	Priority	Symptom	Workaround	Comments	Attachments
Always	Medium	Normal	Incorrect operation	No	If the customer has no special plans for using “read-only” attribute on files in the input directory, the Easiest solution is to remove the attribute once it is detected.	-

Defect Report Fields – Part 1

Software Testing Introduction

TRAINING CENTER

Defect Report Fields – Part 2

Software Testing Introduction

Key defect report fields: general overview

ID	Summary	Description	Steps to reproduce
19	Infinite loop on input file with read-only attribute	If an input file has the “read-only” attribute, the app can not move the processed file into the destination directory: so the app processes the file again and again and thus falls into the infinite loop. **Exp**: the processed file is moved from the input directory to the destination directory. **Act:** the processed data (new file) appears inside the destination directory, but the original file is not deleted from the input directory. **Req:** DS-2.1.	1. Place a valid file (size, type) into the input directory. 2. Set the “read-only” attribute on this file. 3. Start the app. **Bug:** the processing result appears inside the destination dir (and the file is repeatedly updated according to the last write time), but the original file stays inside the input directory.

Reproducibility	Severity	Priority	Symptom	Workaround	Comments	Attachments
Always	Medium	Normal	Incorrect operation	No	If the customer has no special plans for using “read-only” attribute on files in the input directory, the Easiest solution is to remove the attribute once it is detected.	-

Key defect report fields: reproducibility

- Shows if the defect appear each time we follow steps to reproduce (“always”), or if the defect sometimes appears and sometimes doesn’t (“sometimes”).
- Defects with the reproducibility “always” are much more easy to fix.

19		**Req:** DS-2.1.	input directory.

Reproducibility

Always	Medium	Normal	Incorrect operation	No	If the customer has no special plans for using “read-only” attribute on files in the input directory, the Easiest solution is to remove the attribute once it is detected.	-

Key defect report fields: severity

19	Infinite loop on inp with read-only attri

- Shows the damage the defect causes.
- Typical values are:
  - Critical
  - Major
  - Medium
  - Minor

Severity

Always	Medium	Normal	Incorrect operation	No	If the customer has no special plans for using “read-only” attribute on files in the input directory, the Easiest solution is to remove the attribute once it is detected.	-

Key defect report fields: priority

19	Infinite loop with read-or	nto n ir ed , the

- Shows the urgency to fix the defect.
- Typical values are:
  - ASAP
  - High
  - Normal
  - Low

Priority

Always	Medium	Normal	Incorrect operation	No	If the customer has no special plans for using “read-only” attribute on files in the input directory, the Easiest solution is to remove the attribute once it is detected.	-

Key defect report fields: severity vs priority

Severity	**Priority**
Shows, **how dangerous** the defect is	Shows, **how quickly** the defect should be fixed

There may be **any** combination of “severity & priority”

Key defect report fields: symptom

Allows defects classification by typical indication:

- Cosmetic flaw.
- Data corruption/loss.
- Documentation issue.
- Incorrect operation.
- Installation problem.
- Localization issue.
- Missing feature.
- Scalability issue.
- Low performance.
- System crash.
- Unexpected behavior.
- Unfriendly behavior.
- Variance from specs.
- Enhancement.

Symptom

Always	Medium	Normal	Incorrect operation	No	If the customer has no special plans for using "read-only" attribute on files in the input directory, the Easiest solution is to remove the attribute once it is detected.	-

Key defect report fields: workaround

19	Infinite loop on input file with read-only attribute	If an input file has the “read-only” attribute, the app can not move the processed file into the destination directory: so the app processes the file again and again and thus falls into the infinite loop.	1. Place a valid file (size, type) into the input directory. 2. Set the “read-only” attribute on this file. ir ed , the

- Shows, if there is a way to achieve the desired result without being interrupted by the defect.
- Typical values are: “yes”, “no”.

Workaround

Always	Medium	Normal	Incorrect operation	No	If the customer has no special plans for using “read-only” attribute on files in the input directory, the Easiest solution is to remove the attribute once it is detected.	-

Key defect report fields: comments

19	Infinite loop on input file with read-only attribute	If an input file has the "read-only" attribute, the app can not move the processed file into the destination directory: so the app processes the file again and again and thus falls into the infinite loop. Exp: the processed file is moved from the input directory to	1. Place a valid file (size, type) into the input directory. 2. Set the "read-only" attribute on this file. 3. Start the app. ir ed , the

- Optional field.
- May contain any data useful to the process of bug fixing.

Comments

Always	Medium	Normal	Incorrect operation	No	If the customer has no special plans for using "read-only" attribute on files in the input directory, the Easiest solution is to remove the attribute once it is detected.	-

Key defect report fields: attachments

19	Infinite loop on input file with read-only attribute	If an input file has the “read-only” attribute, the app can not move the processed file into the destination directory: so the app processes the file again and again and thus falls into the infinite loop.	1. Place a valid file (size, type) into the input directory. 2. Set the “read-only” attribute on this file. 3. Start the app. ir ed , the

- Optional field.
- May contain any files, screenshots, etc. useful to the process of bug fixing.

Attachments

Always	Medium	Normal	Incorrect operation	No	If the customer has no special plans for using “read-only” attribute on files in the input directory, the Easiest solution is to remove the attribute once it is detected.	-

Useful ideas

Use active voice and simple phrases in steps description

Use objective description in actual/expected results

Write simple, this is not a fiction novel

Use exact names for interface elements

Don’t explain basics

Key defect report fields: general overview

ID	Summary	Description	Steps to reproduce
19	Infinite loop on input file with read-only attribute	If an input file has the “read-only” attribute, the app can not move the processed file into the destination directory: so the app processes the file again and again and thus falls into the infinite loop. **Exp**: the processed file is moved from the input directory to the destination directory. **Act:** the processed data (new file) appears inside the destination directory, but the original file is not deleted from the input directory. **Req:** DS-2.1.	1. Place a valid file (size, type) into the input directory. 2. Set the “read-only” attribute on this file. 3. Start the app. **Bug:** the processing result appears inside the destination dir (and the file is repeatedly updated according to the last write time), but the original file stays inside the input directory.

Reproducibility	Severity	Priority	Symptom	Workaround	Comments	Attachments
Always	Medium	Normal	Incorrect operation	No	If the customer has no special plans for using “read-only” attribute on files in the input directory, the Easiest solution is to remove the attribute once it is detected.	-

Defect Report Fields – Part 2

Software Testing Introduction

TRAINING CENTER

Typical Defect Reporting Mistakes

Software Testing Introduction

TRAINING
CENTER

The defect report is a bad one if…

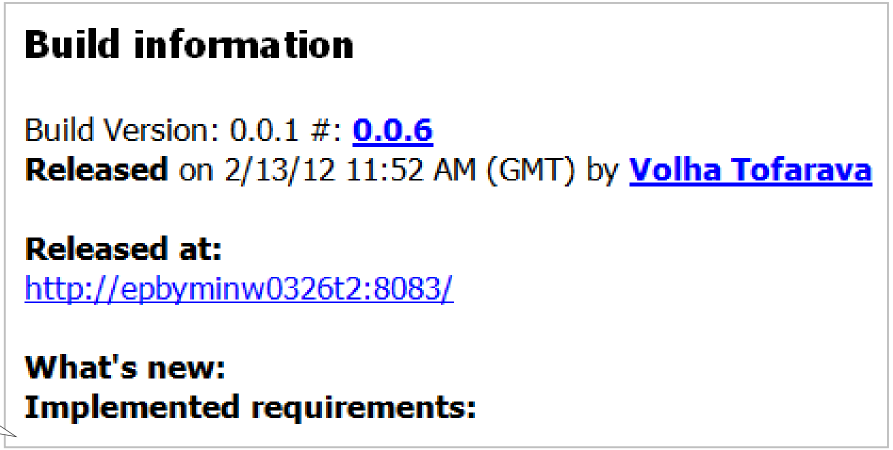
Not enough information to understand and reproduce the defect.

Description
~~The search returns wrong results if some administrative settings are changed.~~

Really!? Such a shame…

The defect report is a bad one if…

The “defect” is found in a functionality not yet marked as “ready for testing”.



Build information

Build Version: 0.0.1 #: **0.0.6**
Released on 2/13/12 11:52 AM (GMT) by **Volha Tofarava**

Released at:
http://epbyminw0326t2:8083/

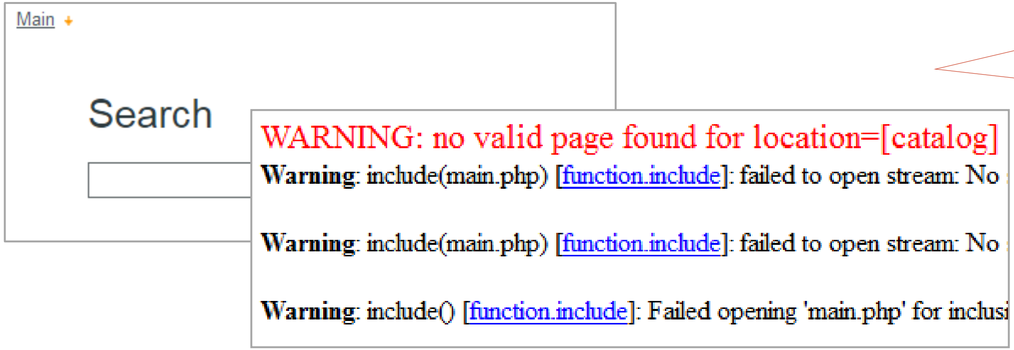
What's new:
Implemented requirements:

Always read build release notes and look to the project management system to be sure the functionality is ready for testing.

The defect report is a bad one if…

Any field contains wrong information (for any reason – copy-paste issues, misunderstanding, inattentiveness, etc…)

Summary
Search page is not available



WARNING: no valid page found for location=[catalog]

Warning: include(main.php) [function.include]: failed to open stream: No

Warning: include(main.php) [function.include]: failed to open stream: No

Warning: include() [function.include]: Failed opening 'main.php' for inclusi

The “Search” page is OK. It’s the “Catalog” page that’s missing.

The defect report is a bad one if…

A defect report contains slang or strong words.

~~Description~~
~~Some s[censored] has happened again to this f[censored] back-redirectors!~~

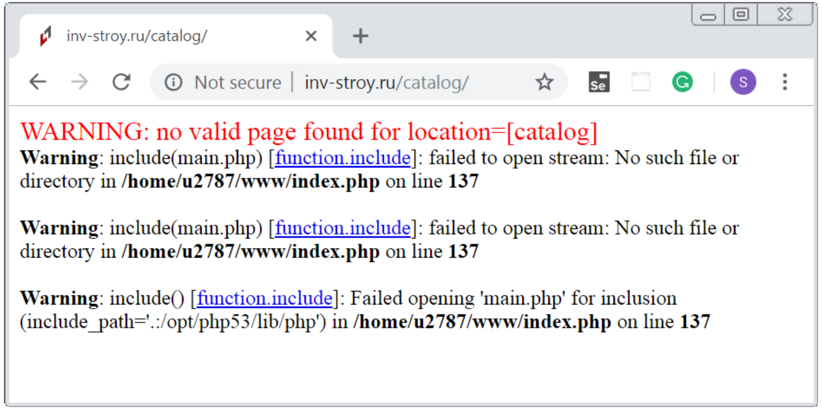
The defect report is a bad one if…

A defect report contains criticism to someone's work.

Description
…. ~~P.S. What a stupid fool may fail with such simple feature implementation?!~~

The defect report is a bad one if…

Some critical details (e.g. the environment specifics) are missing in the report.

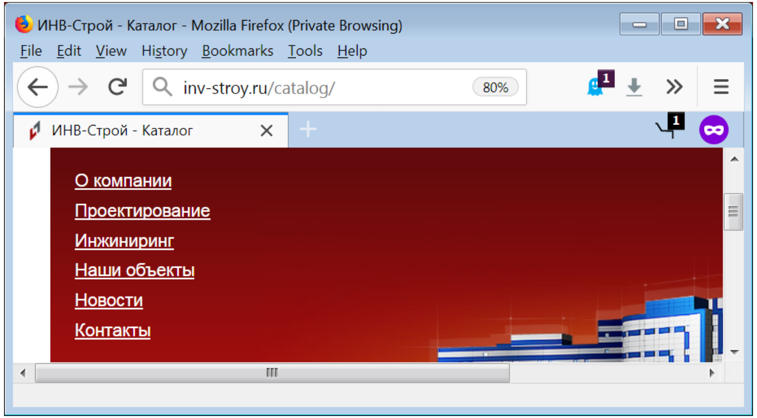


```
WARNING: no valid page found for location=[catalog]
Warning: include(main.php) [function.include]: failed to open stream: No such file or
directory in /home/u2787/www/index.php on line 137

Warning: include(main.php) [function.include]: failed to open stream: No such file or
directory in /home/u2787/www/index.php on line 137

Warning: include() [function.include]: Failed opening 'main.php' for inclusion
(include_path='.:/opt/php53/lib/php') in /home/u2787/www/index.php on line 137
```

W7, Chr, EN



W10, FF, RU

The defect report is a bad one if…

A defect report has inappropriate severity/priority values.

Summary	**Severity**	**Priority**
Each third restart leads to BSOD and loss of ALL user data.	Minor	Low

Bad joke ☹

The defect report is a bad one if…

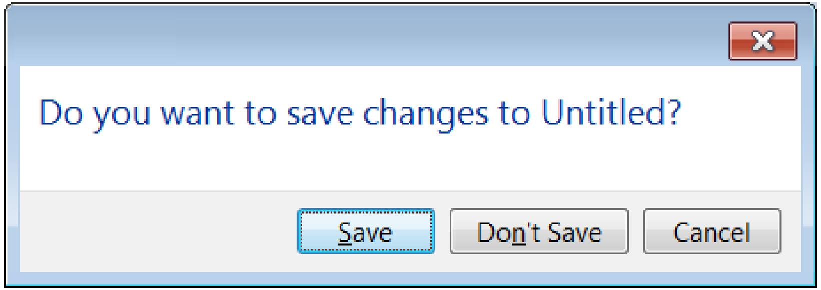
A defect report is written messy, illiterately, confusingly.

Summary	Description	Steps to reproduce
~~No save DB.~~		~~1. Save DB.~~ ~~2. Change DB.~~ ~~3. Create DB.~~ ~~4. Save no.~~

The defect report is a bad one if…

Necessary logs, screenshots, etc. are missing from the report.

Summary	Description	STR
The second click on “New” brings up a window with unreadable signs.	With a document loaded, the second click on “New” menu item brings up a window with unreadable signs (nothing like letters).	1. Open a document. 2. Click “New”. 3. Click “New” again.

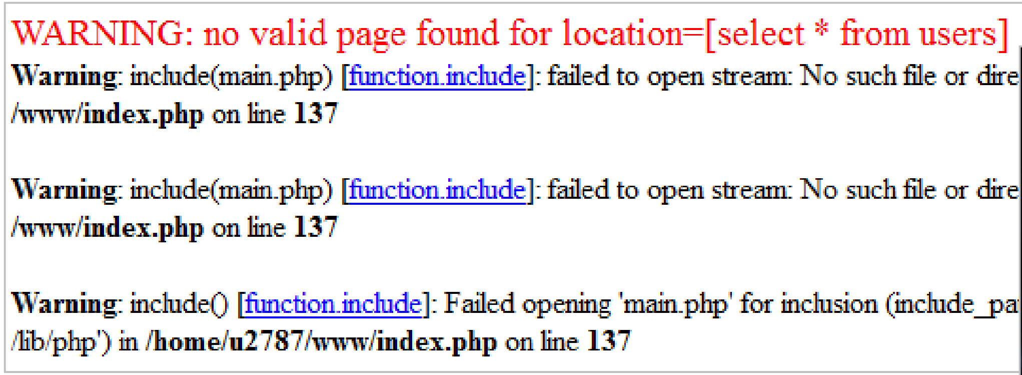


Do you want to save changes to Untitled?

But the developer sees this window. So, where is the bug?!

The defect report is a bad one if…

The tester didn’t manage to convince the team about the defect consequences.



```
WARNING: no valid page found for location=[select * from users]
Warning: include(main.php) [function.include]: failed to open stream: No such file or dire
/www/index.php on line 137

Warning: include(main.php) [function.include]: failed to open stream: No such file or dire
/www/index.php on line 137

Warning: include() [function.include]: Failed opening 'main.php' for inclusion (include_pa
/lib/php') in /home/u2787/www/index.php on line 137
```

Is it trifle?

They don’t think so…

Conclusion

Most mistakes come from the inattentiveness. So be careful, re-read your defect report.

Try to see the technical background, the real cause of the defect. Don’t stop at the interface behavior only.

The more “tricky” defect you found, the more time it may take to deal with. But it saves much more time in the future.

<epam>

Typical Defect Reporting Mistakes

Software Testing Introduction

TRAINING
CENTER
<epam>

Defect Reporting Recommendations

Software Testing Introduction

General idea…

All the next points are just variations of one synonymous series

Attentiveness

Thoroughness

Accuracy

Exactness

Particularity

Nicety

Scrupulousness

…

Follow this advice!

Describe the STR with even the smallest details.

STR

1. Clear cookies or create a fresh new user profile.
2. Open **http://pathtoapplication/admin_login/**
3. Enter **“dba_admin” into the “Login”** field.
4. Enter **“globaladmin12” into the “Password”** field.
5. Click the logo at the left top corner of the page.

Bug: there is a redirection to the DBMS main administration page without additional authentication.

Follow this advice!

Add details not only to STR, but to all other fields

Description
Wrong calculation of the total price of the order in the basket. It seems like the last item in the list is processed twice. **Act**: the “Total” field contains the prices sum + the price of the last item. **Exp:** the “Total” field contains the exact sum of all prices of all items in the list. **Req:** R892.34b/73.

Follow this advice!

Write down every detail, don’t rely on the “esoteric” knowledge.

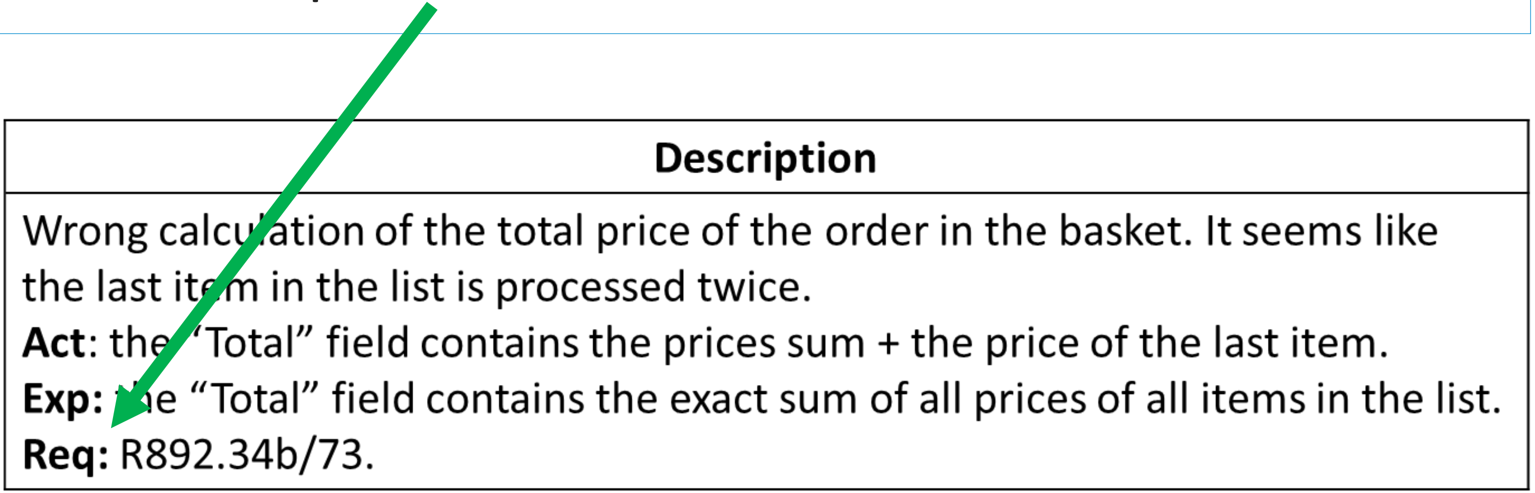
STR

1. Clear cookies or create a fresh new user profile.
2. Open http://pathtoapplication/admin_login/
3. **Change the current language to any different from default.**
4. Authenticate with your domain credentials.
5. Perform a search for any dismissed employee (**search has to return 2+ results**).

Bug: the first match doesn’t contain a photo.

Follow this advice!

Reference to the requirement the defect violates.



Description
Wrong calculation of the total price of the order in the basket. It seems like the last item in the list is processed twice. **Act**: the "Total" field contains the prices sum + the price of the last item. **Exp:** the "Total" field contains the exact sum of all prices of all items in the list. **Req:** R892.34b/73.

Follow this advice!

Provide key details explicitly!

Description
After re-authentication **caused by logout after 10 minutes timeout**, the each-minute auto saving of current document stops working. **Act**: the app either doesn’t auto save the document, or the auto saving process fails with no error messages. **Exp:** the app automatically saves the current document each minute, corresponding hint appears for two seconds at the right top corner of the main app window. **Req:** R1752.2a/4.

Follow this advice!

Even if your BTS doesn’t provide such feature, name the environment the defect found within.

Summary
[XP, IE6, ru] JQuery fails to initialize.

Follow this advice!

Don’t be emotional. Even if you consider the defect to be someone’s epic fail, just provide details, not your opinion.

Summary
The license agreement main page contains 72 typos.

Follow this advice!

Write separate defect report for each found defect. Don’t combine several defects into one defect report.

~~**Summary**~~
~~The catalog map export to JPG causes BSOD.~~ ~~The search doesn’t take into account goods description.~~

But! If you are **absolutely** sure you found one defect with several indications, you can write one defect report and provide the list of indications in the Description field.

Follow this advice!

Make the root cause analysis. Try to find the real cause of the defect. Don’t stop with only system behavior description.

Comments
According to the symptoms, the performance falls to almost zero once the DB size reaches 70-80% of the RAM. It seems that the app tries to load all the DB into the RAM, while it really needs only 3-5% of records.

Follow this advice!

If you have enough technical experience, provide a recommendation on how to fix the defect.

Comments
… We may try optimizing search queries, add caching or at least removing “old datasets” from RAM. This may help reduce the memory usage and should really help those users who perform a lot of “heavy” queries. Even better: we can try to adjust our DAO-layer not to buffer “ahead data”. See the description of the similar problem here: http://habr.com/articles/art97234/

Follow this advice!

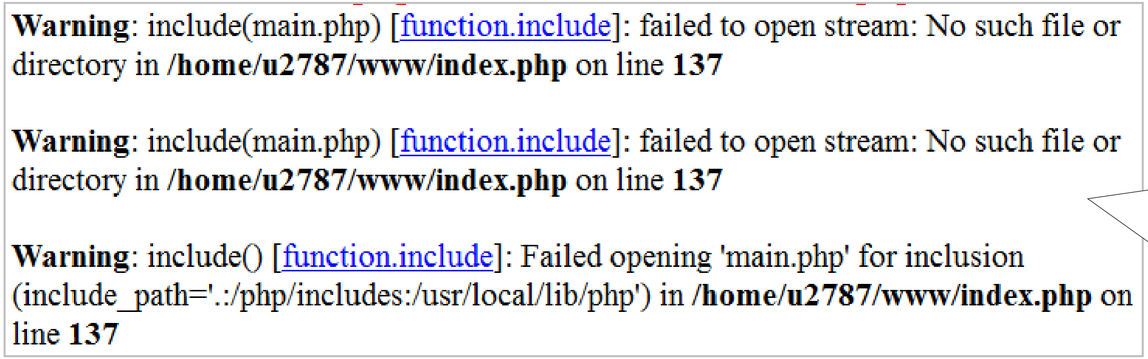
Write down the defect report immediately once you found it and performed all the necessary actions (like root cause analysis, checking the BTS for duplicates and so on).

The less time has passed, the more details you remember.

The sooner the defect report appears, the sooner someone starts working on the defect.

Follow this advice!

Analyze the most critical consequences of the defect. This information will not only help with Severity and Priority fields, but may lead you to some useful ideas on additional tests.



```
Warning: include(main.php) [function.include]: failed to open stream: No such file or
directory in /home/u2787/www/index.php on line 137

Warning: include(main.php) [function.include]: failed to open stream: No such file or
directory in /home/u2787/www/index.php on line 137

Warning: include() [function.include]: Failed opening 'main.php' for inclusion
(include_path='.:/php/includes:/usr/local/lib/php') in /home/u2787/www/index.php on
line 137
```

E.g., this is not just “page not found” issue, this is also security issue and an indication of web app engine misconfiguration.

Follow this advice!

It’s likely that you yourself will verify that the defect is fixed. So writing a good defect report now saves the future you a lot of time and effort.

Follow this advice!

Use spellchecking…

And remember about…

Attentiveness

Particularity

Thoroughness

Nicety

Accuracy

Scrupulousness

Exactness

…

<epam>

Defect Reporting Recommendations

Software Testing Introduction

TRAINING
CENTER
<epam>

«File Converter» Project

Defect Reports Sample
SAMPLE

This is **NOT** a part of project documentation! This is just a way to demonstrate some defect reports.

Defect report sample 1:

ID	Summary	Descriptions	Steps to reproduce	Reproducibility	Severity	Priority	Symptom	Workaround	Comments	Attachments
19	Infinite loop on input file with read-only attribute	If an input file has the “read-only” attribute, the app can not move the processed file into the destination directory: so the app processes the file again and again and thus falls into the infinite loop. **Exp**: the processed file is moved from the input directory to the destination directory. **Act:** the processed data (new file) appears inside the destination directory, but the original file is not deleted from the input directory. **Req:** DS-2.1.	1. Place a valid file (size, type) into the input directory. 2. Set the “read-only” attribute on this file.. 3. Start the app. **Bug:** the processing result appears inside the destination dir (and the file is repeatedly updated according to the last write time), but the original file stays inside the input directory.	Always	Medium	Normal	Incorrect operation	No	If the customer has no special plans for using “read-only” attribute on files in the input directory, the Easiest solution is to remove the attribute once it is detected.	-

Defect report sample 2:

ID	Summary	Descriptions	Steps to reproduce
27	Non-empty subdirectory in SOURCE_DIR is deleted w/o any warning	The app doesn't produce any warning and deletes any non-empty (empty as well) subdirectory in SOURCE_DIR directory. **Exp:** if the app detects a non-empty subdirectory inside SOURCE_DIR, it produces the error message and stop. The message should be: "Non-empty subfolder [name] in SOURCE_DIR folder detected. Remove it manually or restart application with --force_file_operations key to remove automatically." **Act:** any non-empty subdirectory (along with all its contents) in SOURCE_DIR is deleted without any warning. **Req:** DS-8.5.12.	1. Create a directory with any valid name (e.g. TEST). Create a subdirectory (e.g. BUG) inside it and place some files inside this subdirectory. 2. Start the app with the SOURCE_DIR parameter pointing to the TEST directory (see the step 1), e.g.: "php file_converter.phar c:\TEST c:\OUT". **Bug:** the subdirectory BUG is deleted with all its contents.

Reproducibility	Severity	Priority	Symptom	Workaround	Comments	Attachments
Always	Major	Normal	Data loss	No		-

Defect report sample 3:

ID	Summary	Descriptions	Steps to reproduce
41	File processing stops after 0.5 hour of work	After something about 30 minutes of work the app stops reacting to the new valid files inside the SOURCE_DIR. **Exp:** the app continues to process the new valid files in the SOURCE_DIR until it is stopped by an appropriate command. **Act:** the app still utilizes some system resources (CPU, RAM, drives) but does not react to any new valid file inside the SOURCE_DIR. **Req:** UR-1.	1. Start the app and let it work for at least 30 minutes. 2. Put a valid file inside the SOURCE_DIR. **Bug:** the app does not process the file.

Reproducibility	Severity	Priority	Symptom	Workaround	Comments	Attachments
Always	Critical	Normal	System crash	No	It looks like some descriptors of file system objects are closed automatically due to some timeout. This behavior may have the same cause as in bug 38 (once started, the app does not see that SOURCE_DIR and/or DESTINATION_DIR are deleted).	-

Defect report sample 4:

ID	Summary	Descriptions	Steps to reproduce
2352	Multiple instances crash on competition for the SOURCE_DIR	With 2+ instances of the application running with the same input directory configured, they compete for files in this directory that leads to multiple file operation failures and (finally) to crash of some instances. **Exp:** all instances work (not crashes); if a file operation fails, the affected instance makes a log record and retries the operation. **Act:** the affected instance crashes. **Req:** UR-12.	1. Start several (3-5-7+) instances of the app with **the same** valid parameters. 2. Start putting valid files into the SOURCE_DIR. **Bug:** one by one most instances crash.

Reproducibility	Severity	Priority	Symptom	Workaround	Comments	Attachments
Always	Critical	Normal	System crash	No	It looks like some semaphore or synchronization mechanism is either broken or do not take the SOURCE_DIR into account. Or some file operation failure handler does not work as it has to.	-