ISBSG
Delivering IT Confidence

The global and independent source of data and analysis for the IT industry

Short paper

Analysis of Defects

Published by the International Software Benchmarking Standards Group

2021

Introduction

As the ISBSG repository contains more data of projects carried out in an agile way of working, analysis of differences between traditional projects and agile projects becomes more significant. The ISBSG collects industry data, where output is measured using ISO/IEC standardized and therefore objective, repeatable, auditable methods, such as Nesma, IFPUG and COSMIC function points. Typical key metrics based on function points are:

- Project Delivery Rate (PDR)[1]: Hours spent per function point
- Cost efficiency: Cost (or Price) per function point
- Quality: Defects per function point (in test and/or 1st month of production)
- Speed: Function points delivered per calendar month.

The ISBSG repository 'New Developments & Enhancements' contains thousands of completed projects for which these metrics are calculated, enabling organizations to use this industry data for fact-based understanding and decision making. In this short paper, the difference in productivity between traditional and agile projects is analyzed.

Defects

According to Wikipedia, a software bug (= defect) is an error, flaw or fault in a computer program or system that causes it to produce an incorrect or unexpected result, or to behave in unintended ways. The process of finding and fixing bugs is termed "debugging". This often uses formal techniques or tools to pinpoint bugs. Since the 1950s, some computer systems have been designed to also deter, detect or auto-correct various computer bugs during operations.

Software Defects, as defined by ISBSG, are the defects found during the first month after implementation of the project or release. Usually there is also a typology indicating the severity of the defects, like minor / major / extreme defects. An important metric in the ISBSG repository is the Defect Density metric, which is defined as Defects per 1000 FP.

Defect Density is calculated as Total Defects Delivered * 1000 / Functional Size

[1] The PDR is actually the inverse of the universal concept of Productivity (output/input) as it is easier to process for human minds, which usually struggles with metrics with many decimals.

An example: In a functional release 100 FP were created and within 1 month after implementation 5 defects were found, the Defect Density would be calculated: 5 * 1000 / 100 = 50 per function point.

The Defect Density says something about process quality and test effectiveness. The higher this number, the more effort goes into corrective maintenance activities, which increases the total cost of ownership (TCO) of the application.

In this short paper, the Defect Density is analyzed: through time and by size category.

Defect Density versus Productivity in the ISBSG repository.

In this short paper, we are looking at the Defect Density of projects per Size Category and through time.

For the traditional project dataset, the following selection criteria were applied:

- Data Quality Rating: A or B
- Primary Programming Language: .Net or Java
- Count approach: Nesma or IFPUG 4+
- Defect Density <> Blank

This results in 91 data points. As usually only organizations with a high maturity level are collecting defects after implementation, the number of data points is quite low.

In the following figure, the distribution of the Defect Density per Size Category of this data set is shown.

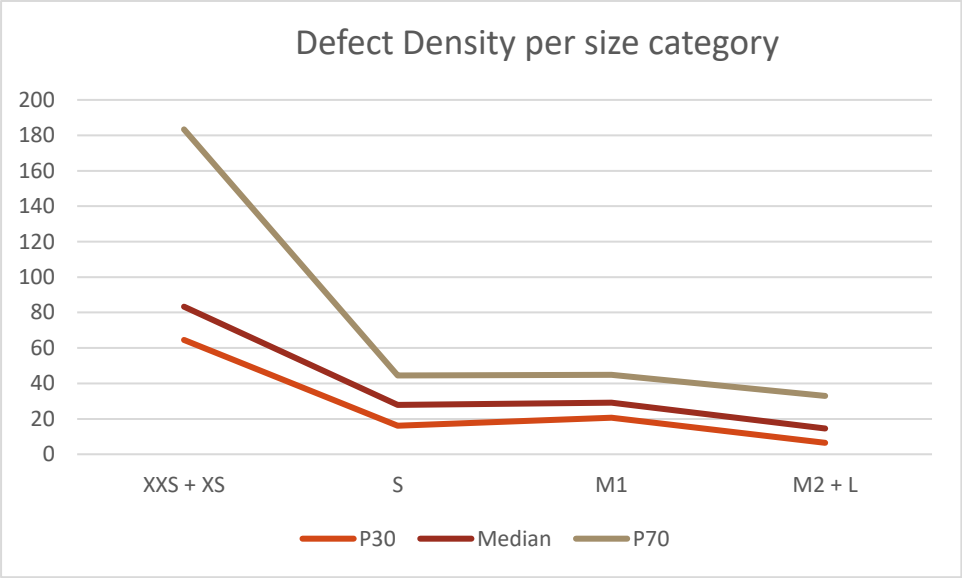

Figure 1 Defect Density for Java and .Net projects per size category in the 2021 ISBSG D&E repository.

Size	Relative Size	Functional Size (FP)
XXS	Extra Extra Small	=> 0 and <10
XS	Extra Small	=> 10 and <30
S	Small	=> 30 and <100
M1	Medium 1	=> 100 and <300
M2	Medium 2	=> 300 and <1,000
L	Large	=> 1,000 and < 3,000

The figure shows that the smaller the project, the higher the Defect Density. Obviously, the absolute number of defects gets higher as the size gets larger.

When we take a look at the Defect Density through time, we observe first of all that the number of data submissions with defect data was rising until 2015 but is declining ever since. For the last 5 years, only 6 projects with defect data were submitted.

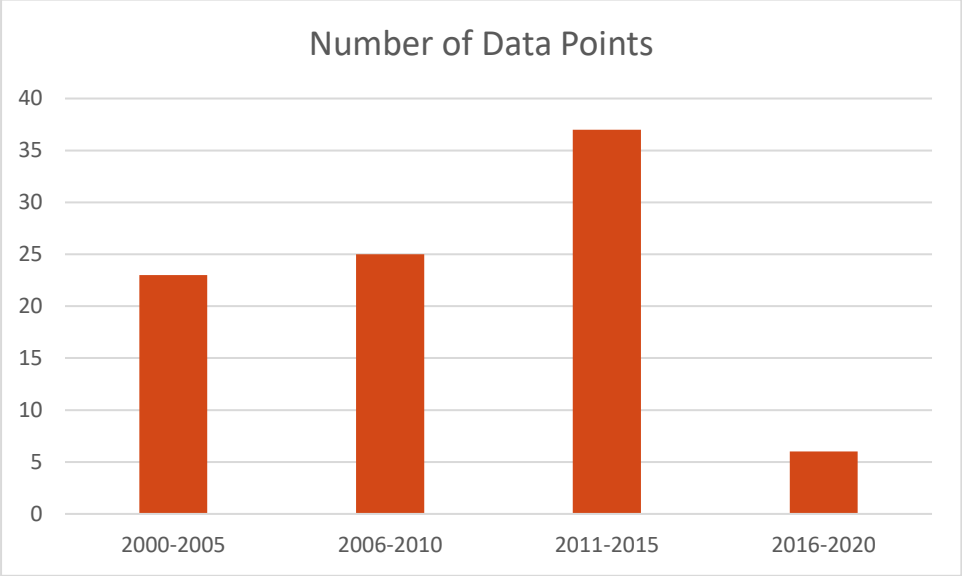

Table 1: Defect Density for Java and .Net projects in the 2021 ISBSG D&E repository – the number of data points per time period.

It seems that fewer organizations have robust metrics in place regarding the number of defects that are delivered after implementation. A similar observation can be made in practice when looking at many agile teams. Defects during tests are often resolved on the spot, without registering them into a defect tracking tool.

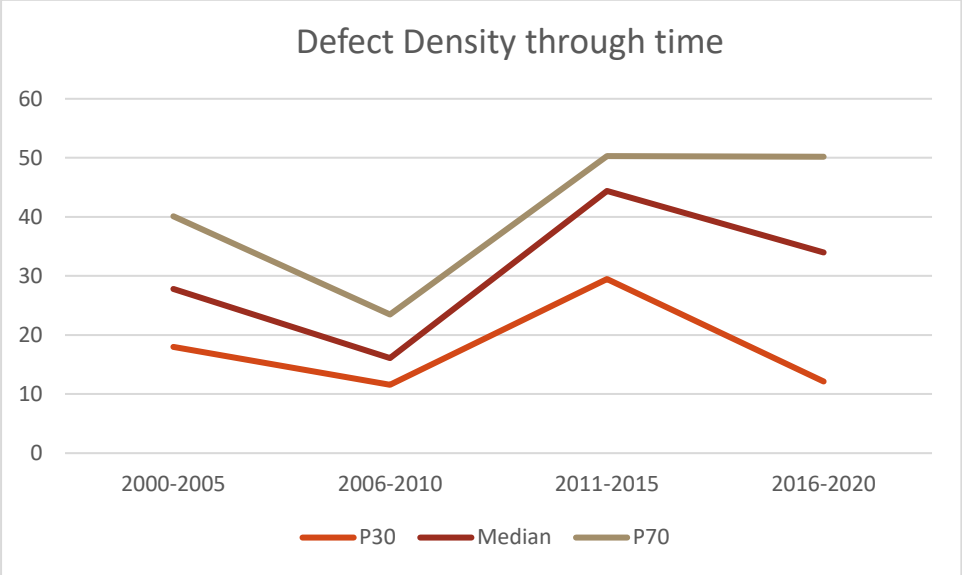

Figure 2 Defect Density for Java and .Net projects in the 2021 ISBSG D&E repository through time.

The graph shows a somewhat erratic line. The small number of data points in the last time period may the cause for that.

Conclusion

Defect Density is an important metric which says something about the effectiveness of the test process. The Defect Density seems to be better when the project gets larger. When looking at the Defect Density through time, it's hard to make a conclusion as the number of project submissions containing defect data is decreasing, resulting in only 6 data points for the period 2016-2020. The fact that submissions of projects developed in an agile way of working is predominant nowadays, and in these type of projects defects are less likely to be registered, may be one of the reasons for this.

If you wish to do your own analysis, or if you are interested to use the ISBSG data for Cost estimation, benchmarking, performance measurement, procurement, etc., please subscribe to the data here: https://www.isbsg.org/project-data/

The International Software Benchmarking Standards Group (ISBSG)

The ISBSG is a not-for-profit organization founded in 1997 by a group of national software metrics associations. Their aim was to promote the use of IT industry data to improve software processes and products.

ISBSG is an independent international organization that collects and provides industry data of software development projects and maintenance & support activities in order to help all organizations (commercial and government, suppliers and customers) in the software industry to understand and to improve their performance and decision making. ISBSG sets the standards of software data collection, software data analysis and software project benchmarking processes and is considered to be the international thought leader in these practices.

The ISBSG mission is to support commercial and public organizations to improve the estimation, planning, control and management of IT software projects and/or maintenance and support contracts.

To achieve this:

ISBSG maintains and grows 2 repositories of IT software development/maintenance & support data. This data originates from trusted, international IT organizations and can be obtained for a modest fee from the website www.isbsg.org/project-data/

Help us to collect data

ISBSG is always looking for new data. In return for your data submission, we issue a free benchmark report that shows the performance in your project or contract against relevant industry peers.

Please submit your data through one of the forms listed on http://isbsg.org/submit-data/

A specific Agile/Scrum data collections questionnaire can be downloaded here: https://cutt.ly/4vnuXVT

Partners

This page will help you to find an ISBSG partner in your country:

http://isbsg.org/meet-isbsg-partners/