Density of Defect

Uthaya Vasanthan S, Karthikeyan B

Abstract-- The main objective of this project is to come out with a new and effective idea for measuring the quality of the software (software quality metrics). The existing product quality metrics which is a subset of the software quality metrics focus on measuring the quality by MTTF [Mean Time To Failure] and DD [DEFECT DENSITY]. We bring in a new idea called the "DENSITY OF DEFECT", stressing that quality of the product can be better judged by measuring the DENSITY of the identified defect, proving that merely the number of defects will not be an effective parameter in quality estimation as stated in DEFECT DENSITY. This project's scope will also include how the density of defect idea can be effective enough in measuring not only the quality but also in reducing the effort of identifying and correcting the individual defect.

Keywords- DD--Defect Density. D(D) -- Density of Defect. LOC -- Lines Of Code.

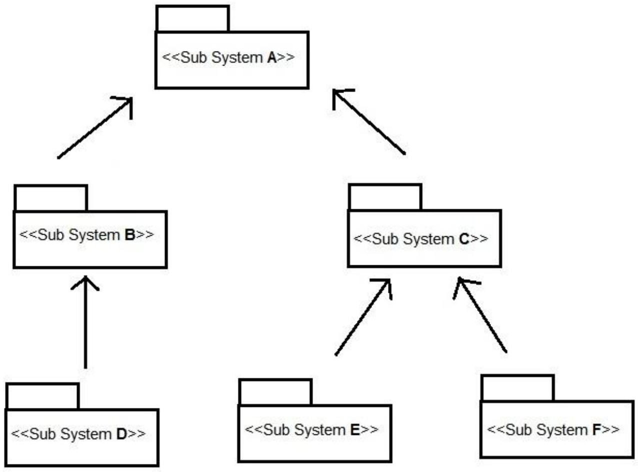
I. INTRODUCTION

Software project metrics uses defect density to measure the quality of software. But when it comes to accuracy the defect density which mainly focuses on just the number of defects found in a system irrespective of the type, nature and location (i.e. more dependent Sub-System or less dependent Sub-System) in which they occur cannot be an effective quality measure. To overcome these shortfalls of the defect density approach we have coined a new term called the "**density of defect**" which mainly concentrates on the location in which a particular defects occur (i.e. more dependent Sub-System or less dependent Sub-System). According to D (D) the defects found at a more dependent Sub-System is given the highest level of importance. This would be a great parameter in measuring the exact quality. For instance you find two defects in your system when you track one defect arise from a Sub-System which is highly dependent by other Sub-Systems and other defect from a least dependent Sub-System. According to DD both these defects are not differentiated they are treated in the same way. But will this be a valid measure of the quality of the software. These questions will be resolved in our D (D).

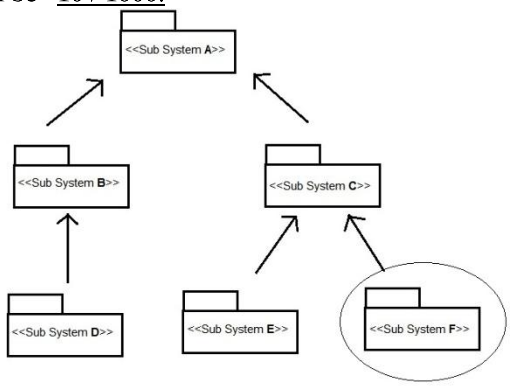
II. PROBLEM DESCRIPTION

Let us take some scenarios and discuss in detail about the problem that we have in DD. DD can be calculated as
DD= Number of defects / LOC
SENARIO 1:

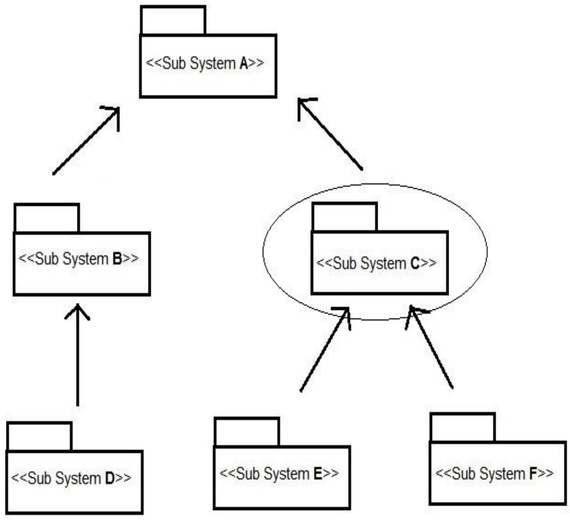
- Let us consider 6 Sub-Systems, namely A,B,C,D,E & F, to calculate the Defect Density, taking that all the 6 Sub-System constitute 1000 LOC.



- Assume that Sub-System **"F"** is tracked to have 10 defects and all other Sub-System with no defects, then the DD will be 10 / 1000.



SENARIO 2:
Assume that Sub-System **"C"** is tracked to have 10 defects and all other Sub-System with no defects, and then the DD will be 10 / 1000 .



Comparing scenario 1 and scenario 2:
DD in scenario 1 is 10 / 1000.
DD inscenario2 is 10 / 1000.
Are they same?

Manuscript Received on July, 2013.

Uthaya Vasanthan.S, Student M.S III Year, SITE School, VIT University, Vellore, Tamil Nadu, India.
Karthikeyan.B, Student M.S III Year, SITE School, VIT University, Vellore, Tamil Nadu, India.

- If "F" which is a least dependent Sub-System given the same importance to that of the more dependent Sub-System.
- Can you give the defects identified in the above two scenarios in the same frame.
- Can all defects be treated similar, irrespective of the Sub-System in which they occur?

Answer:

No, they are not same, as stated by Defect Density (DD). This is because the Sub-System "**F**" is not dependent by other Sub-Systems therefore the defects tracked to that Sub-System is localized (i.e. no chance of migration).

But on the other hand the Sub-System "**C**" is dependent by two other Sub-Systems namely Sub-System" **E**" and Sub-System "**F**". The chance of defect migration is higher. A defect input from Sub-System "**C**" will naturally result in a defect output in its dependent Sub-Systems (i.e. due to migration). Even in situations where the defects do not migrate the risk of having a defect at a more dependent Sub-System has to be considered.

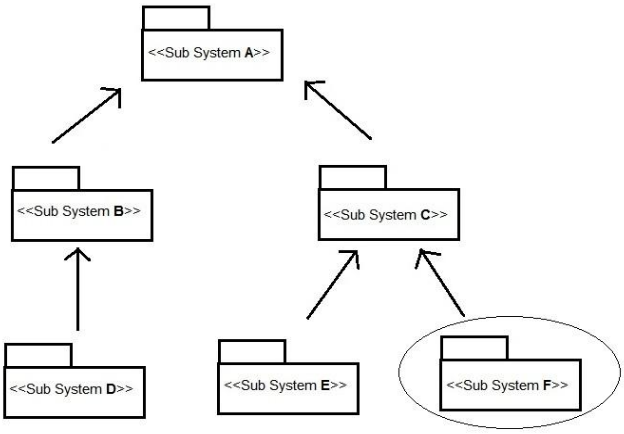
III. DENSITY OF DEFECT [D (D)]

Any defect identified can be mapped to the responsible Sub-System. Asystem is a collection of dependent Sub-Systems. A defect tracked to a more dependent Sub-System is said to have a high impact when compared, with least dependent Sub-System (i.e., the density of defect found in a more dependent Sub-System will be higher). To calculate the density of defect, we need to adapt to a particular strategy. The strategy that will be relevant or suitable for calculating the density of defect will be "BOTTOM – UP" approach.

The child node can be tracked to its parent node. Similarly any defect found in the child Sub-System has to be tracked back to its parent Sub-System, until the last parent of that particular path. This enables us to identify the density of defect. Therefore D (D) can be calculated as follows

D (D) = number of defects in the path / LOC of the path

Consider Scenario 1



Any defect in the Sub-System F, has to be tracked back to its parent C, which in turn is tracked back to its parent that is Sub-System A. We don't consider the entire system, as a defect identified in Sub-System F will not have any impact on Sub-System D, B, and E as they are not dependent.

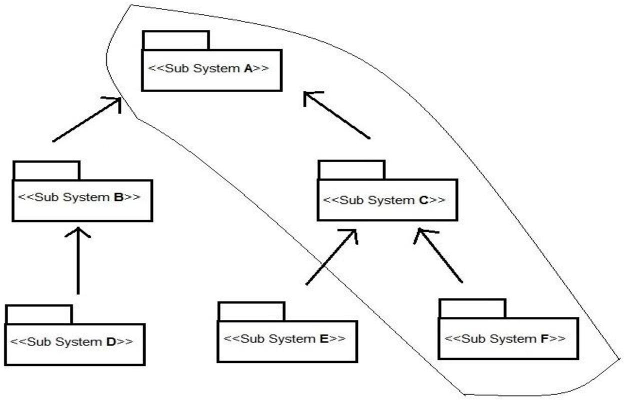
IV. DENSITY OF DEFECT IN PRACTICE

Let us consider 6 Sub-Systems, namely A,B,C,D,E & F, to calculate the Defect Density, taking that all the 6 Sub-System constitute 1000 LOC. The Sub-Systems have the following LOC.

A - 100 LOC
B - 100 LOC
C - 150 LOC
D - 250 LOC
E - 100 LOC
F - 300 LOC

SCENARIO 1:

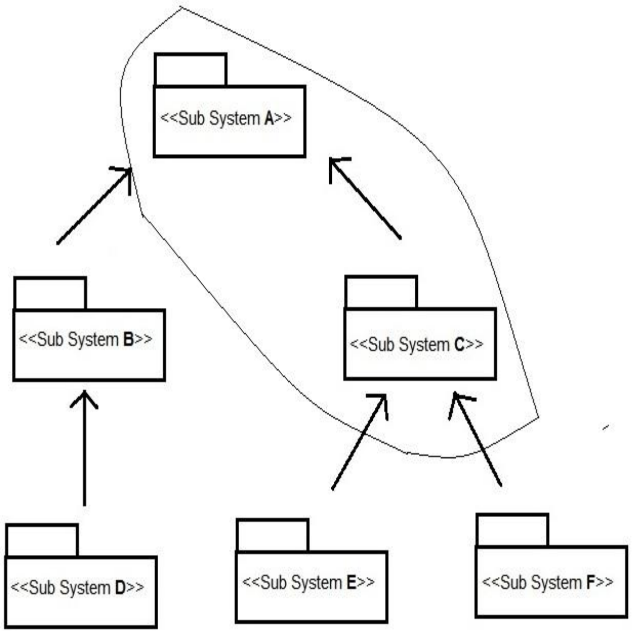
- Assume that Sub-System "**F**" has 10 defects and all other Sub-Systems with no defects and that particular path has 550 LOC.



Then D (D) at F is <u>10 + 0 + 0 / 550 = 10 / 550.</u>

SCENARIO 2:

- Assume that Sub-System "**C**" has 10 defects and all other Sub-Systems with no defects and that particular path has 250 LOC.



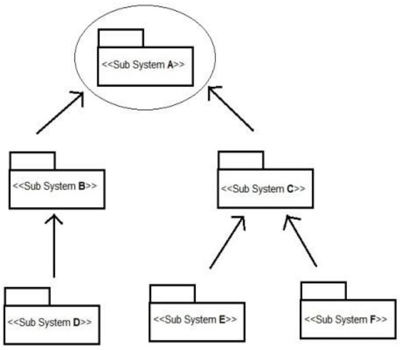
Then D (D) at C is <u>10 + 0 / 250 = 10 / 250 .</u>

Now, you can clearly see the impact or how dense a defect at a particular Sub-System will be i.e., High dependent Sub-System will always have a high density, if a defect is tracked to it.

SCENARIO 3:

- Assume that Sub-System "**A**" has 10 defects and all other Sub-Systems with no defects, if a defect is tracked to a Sub-System "**A**", which is highly dependent, you could see the impact of defect.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication



LOC of the path here is the LOC of the Sub-System itself (assuming LOC to be 100).

Density of any defect at A will be 10/100.

Comparing Scenario 1, Scenario 2 & Scenario 3, We get

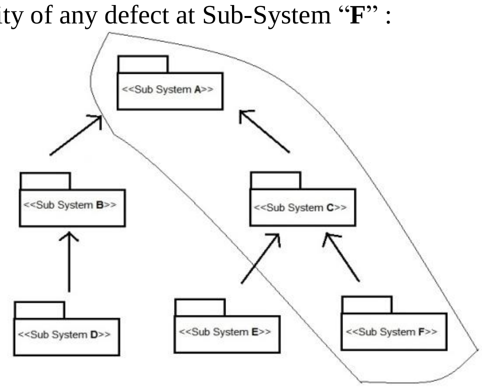
Scenario 1		Scenario 2		Scenario 3
10 / 550	<	10 / 250	<	10 / 100
Sub – System F		Sub – System C		Sub – System A

It is clear that, any defect at A will be denser when compared to the previous cases 1 and 2.

Detailed scenario:

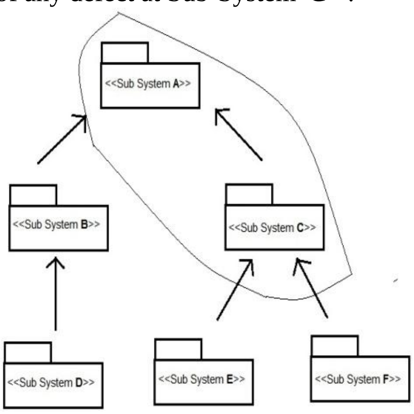
Assume that the no. of defects with the
Sub-System A = 3.
Sub-System B = 2.
Sub-System C = 5.

- Density of any defect at Sub-System "**F**" :



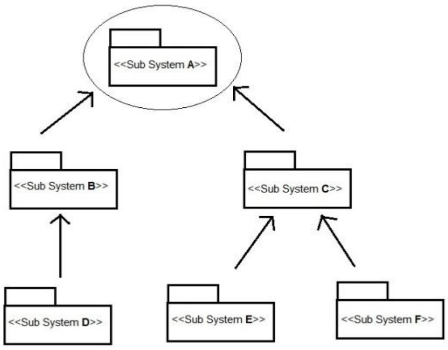
D (D): (5 + 2 + 3) / 550 = 10 / 550 .

- Density of any defect at Sub-System"**C"** :



D (D): (2 + 3) / 250 = 5 / 250 .

- Density of any defect at Sub-System "**A**" :



D (D): 3 / 100 .

V. ADVANTAGES

An exact measure of the quality of the system developed (i.e., rating of the system based on the occurrence rather than the number of calculations) can prioritize the defects based on the Sub-Systems they occur, for the above scenario the defect correction priority would be as follows.

Sub-System	Priority number	Dependent Sub-Systems
A	1	B,C
B	3	D
C	2	E,F
D	4	---
E	4	---
F	4	---

Note: This decision is taken based on their dependency relation.

We could visualize the impact of corrected defect i.e., if a defect at Sub-System "**A**" gets corrected according to the priority, then the other defect related to that particular defect will be resolved, which can be proved by calculating Density of Defect.

VI. CONCLUSION

The D (D) method will thus be a handful tool in measuring the exact quality of any system developed. Its scope extends beyond calculating the quality (i.e. can be used in prioritizing the defects and for better rework practices). It also gives a clear idea about the impact of a defect. You can now realise how dangerous a defect at a more dependent Sub-System would be. This method will also fetch a vital conclusion that should be made mandatory in our system design. The Sub-Systems should be designed in such a way that it has high degree of coherence and very less degree of coupling (i.e. do not create a system with more dependent Sub-Systems). Whenever the degree of coupling of a system is high, there are every chances of it affecting the quality of the system.

REFERENCES

1. Metrics & models in Software Quality Engineering [Stephen- H kan] 2nd edition.
2. Measuring Software Quality [Richard walts].
3. Software Quality, theory & management 2nd edition [Alan C Gillies].
4. Software Engineering practioner's approach [Roger.S.Pressman].
5. Software metrics [C.Ravindranath Pandian].