

UNIT 5

SOFTWARE QUALITY AND METRICS

Need of Object Oriented Software Estimation

The Object-Oriented Project Size Estimation (Oopsize) technique uses the initial estimates of B1 and B2 **to predict how much time is required to design, code and test an object**. The objects can be described in a Rational Rose class model, for instance.

The calculation of **test estimation techniques** is based on:

- Past Data/Past experience
- Available documents/Knowledge
- Assumptions
- Calculated risks

Software Estimation Techniques

There are different Software **Testing Estimation** Techniques which can be used for estimating a task.

- 1) Delphi Technique
- 2) Work Breakdown Structure (WBS)
- 3) Three Point Estimation
- 4) Functional Point Method

Disadvantages of **Software Estimation Techniques**:

- Due to hidden factors can be over or under estimated
- Not really accurate
- It is based on thinking
- Involved Risk
- May give false result
- Bare to losing
- Sometimes cannot trust in estimate

Object Oriented Metrics in Software Engineering

These are used to determine success or failure of a person also to quantify the improvements in the software throughout its process. These metrics can be used to reinforce good OO programming technique which lead to more reliable code. Object-oriented software engineering metrics are units of measurement that are used to characterize:

- object-oriented software engineering products, e.g., designs source code, and the test cases.
- object-oriented software engineering processes, e.g., designing and coding.
- object-oriented software engineering people, e.g., productivity of an individual designer.

SOFTWARE MEASUREMENT:

A measurement is a manifestation of the size, quantity, amount or dimension of a particular

attribute of a product or process. Software measurement is a titrate impute of a characteristic of a software product or the software process. It is an authority within software engineering. The software measurement process is defined and governed by ISO

Standard.

Need of Software Measurement:

Software is measured to:

1. Create the quality of the current product or process.
2. Anticipate future qualities of the product or process.
3. Enhance the quality of a product or process.
4. Regulate the state of the project in relation to budget and schedule.

Classification of Software Measurement:

There are 2 types of software measurement:

1. Direct Measurement:

In direct measurement the product, process or thing is measured directly using standard scale.

2. Indirect Measurement:

In indirect measurement the quantity or quality to be measured is measured using related parameter i.e. by use of reference.

Metrics:

A metric is a measurement of the level that any impute belongs to a system product or process.

There are 4 functions related to software metrics:

1. Planning
2. Organizing
3. Controlling
4. Improving

Characteristics of software Metrics:

1. Quantitative:

Metrics must possess quantitative nature. It means metrics can be expressed in values.

2. Understandable:

Metric computation should be easily understood, the method of computing metric should be clearly defined.

3. Applicability:

Metrics should be applicable in the initial phases of development of the software.

4. Repeatable:

The metric values should be same when measured repeatedly and consistent in nature.

5. Economical:

Computation of metrics should be economical.

6. Language Independent:

Metrics should not depend on any programming language.

Classification of Software Metrics:

There are 3 types of software metrics:

1. Product Metrics:

Product metrics are used to evaluate the state of the product, tracing risks and uncovering prospective problem areas. The ability of team to control quality is evaluated.

2. Process Metrics:

Process metrics pay particular attention on enhancing the long term process of the team or

organization.

3. Project Metrics:

The project matrix describes the project characteristic and execution process.

- Number of software developer
 - Staffing pattern over the life cycle of software
- Cost and schedule
- Productivity

