Software Project Management

Lecture # 4

Outline

- Metrics for Software Quality
  - Defect Removal Efficiency
- Integrating Metrics with the Software Process
  - Arguments for software metrics
  - Establishing a baseline
  - Metrics collection, computation and evaluation
- Establishing a software metrics program

Metrics for Software Quality

- Earlier we discussed different measures of software quality
  - Correctness
  - Maintainability
  - Integrity
  - Usability
- We will now discuss a quality metric: defect removal efficiency (DRE)

Defect Removal Efficiency

- DRE is a measure of the filtering ability of the quality assurance and control activities as they are applied through out the process framework.

Defect Removal Efficiency (Contd.)

- When considered for project as a whole, it is defined as
- DRE = E / (E + D)
  - E = no. of errors found before delivery
  - D = no. of defects found after delivery
- Ideal value of DRE is 1
- Realistically D will be greater than 0
- As E increases, DRE begins to approach 1

Defect Removal Efficiency (Contd.)

- If used as a metric that provides an indicator of the filtering ability of the quality assurance and control activities…

  DRE encourages a software project team to institute techniques for finding as many errors as possible before delivery.

Redefining DRE

- DRE can also be applied on each process framework activity and hence find the team's ability to assess errors before they are passed to next activity or software engineering task.
- $DRE = E_i / (E_i + E_{i+1})$
  - E_i = errors in activity i
  - E_{i+1} = errors in activity i+1 that were not discovered in activity i

Arguments for Software Metrics

- Majority of software developers do not collect measures.
- Without measurement it is impossible to tell whether a process is improving or not.
- Measurement is used to establish a process baseline from which improvements can be assessed.

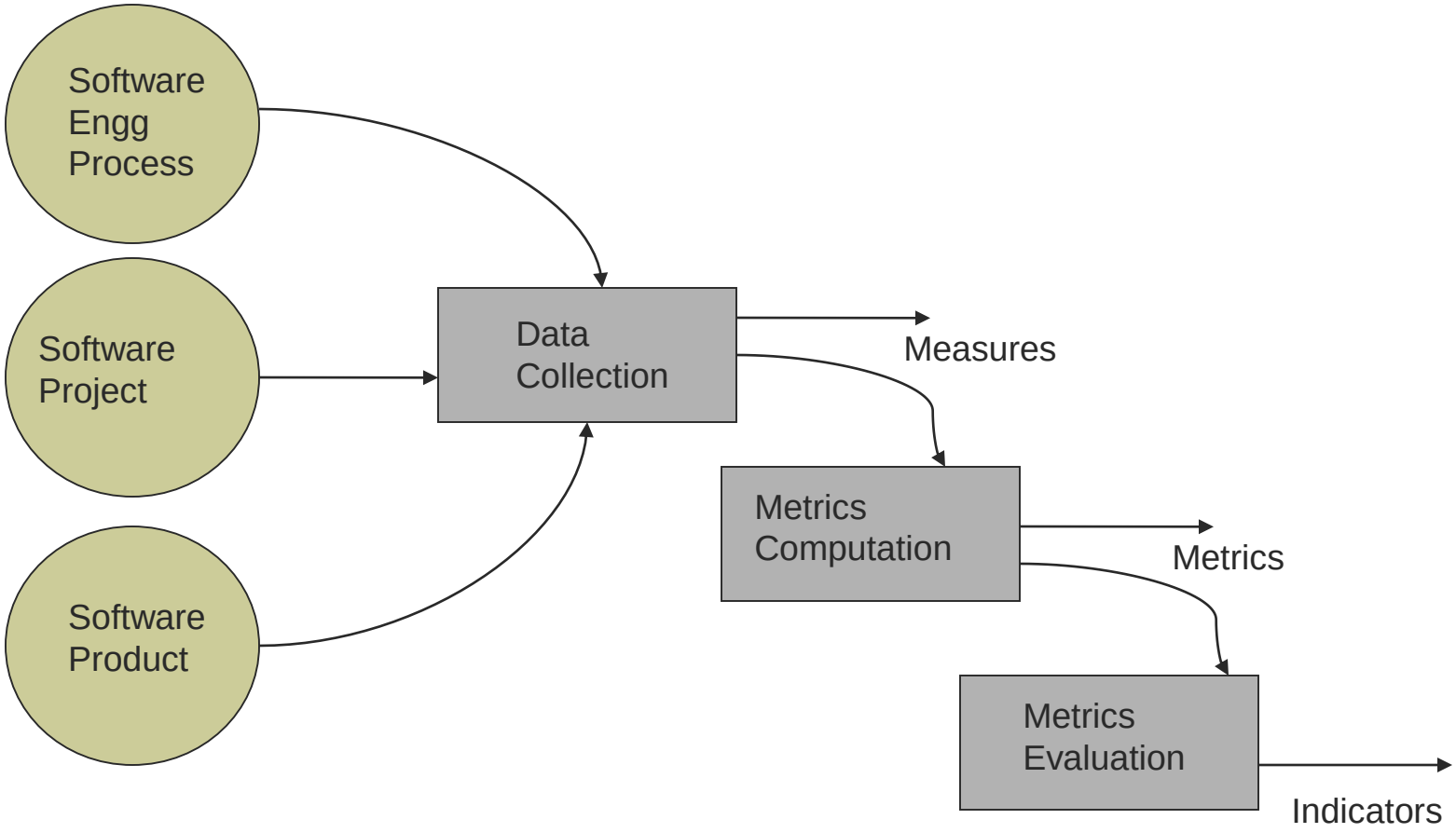
Establishing a Baseline

- The metrics baseline consists of data from past software development projects.
- Baselines can be as simple as data in table or as complex as a complete database having numerous measures and metrics derived from them.
- To be used effectively in process improvement and/or cost estimation, baseline data must have the following attributes.

Establishing a Baseline (Contd.)

- Baseline data attributes:
  - Data must be accurate (no guess work)
  - Data should be collected from as many projects as possible
  - Measures must be consistent across all such projects
  - Baseline data/applications should be similar to the work that is to be estimated

Metrics Collection, Computation and Evaluation



Metrics for Small Organizations (1)

- Most software organizations have fewer than 20 software engineers.
- It is unreasonable to expect that such small organizations will develop comprehensive software metrics program.
- However, it is reasonable to suggest that software organizations of all sizes should measure and then use the resultant metrics to improve their local software process, quality and timelines of product delivery.
- Best advice is to "*Keep it simple*", i.e., choose simple metrics that provide value to the organization and don't require a lot of effort to collect.

Metrics for Small Organizations (2)

- Improvement target: *"to reduce the time to evaluate and implement change requests"*
- A small organization may select the following measures to achieve the target:
  - Time elapsed from the time a request is made until evaluation is complete
  - Effort to perform evaluation
  - Time elapsed from completion of evaluation to assignment of change order to personnel
  - Effort required to make the change
  - Time required to make the change
  - Errors uncovered during work to make the change
  - Defects uncovered after change is released to the customer

Establishing Software Metrics Program (1)

- SEI has suggested the following guidelines for establishing a “goal-driven” Software Metrics Program:
  1. Identify business goal
  2. Identify what you want to know
  3. Identify subgoals
  4. Identify subgoal entities and attributes
  5. Formalize measurement goals
  6. Identify quantifiable questions and indicators related to subgoals

Establishing Software Metrics Program (2)

7. Identify data elements needed to be collected to construct the indicators
8. Define measures to be used and create operational definitions for them
9. Identify actions needed to implement the measures
10. Prepare a plan to implement the measures

Summary

- Measurements result in change in working-culture.
- Measurement helps practitioners and managers to improve the process.
- Data collection, metrics computation and their analysis are the core steps to begin a ‘metrics program’.

THE END