If software development is to be viewed as an engineering discipline, it requires a measurement component that allows us to better understand, evaluate, predict and control the software process and product. This need has generated an area of study, software measurement, that has been evolving slowly over a number of years.

Most of the early work was empirical, i.e., collecting data and trying to postulate models based upon the data [Akiyama71], [Basili&Zelkowitz77], [Belady&Lehman76], [Boehm73], [Endres75], [Walston&Felix77], [Wolverton74]. Some of the work involved the postulation of a metric or even model that tried to explain the nature of the process or product [Boehm,Brown&Lipow76], [Eshoff76], [Fuer&Folkes79], [Halstead77], [McCabe76], [Musa75], [Putnam78]. Measurement was also used for evaluating the hypotheses in controlled experiments, [Basili&Reiter79], [Curtis79], [Gannon&Horning75], [Sackman,Erikson&Graat83] and occasionally was promoted as an integral part of the software development process [Basili&Turner75].

The area of software measurement has continued to progress. The main inroads have been in the scope of measurement, a deeper understanding of the perspectives from which it can be applied, the development of frameworks for its definition and interpretation, the refinement of the measures and the models on which the metrics are based, the automation of the models and measures, and the increased application of measurement in many organizations.

Increased Scope

Historically we began by measuring such things as size (e.g., lines of code), defect counts, and resource usage (e.g. effort and computer time). We now measure many aspects of the process and product and are working to understand the relationship between the two in a variety of environments.

Perspectives

Measurement was originally performed from the perspective of the manager trying to control the development, or the organization trying to understand the quality and cost of the software. We are now trying to use measurement to better understand the process, product and environment, evaluate new technologies, motivate software development and evolution, and improve it.

Frameworks

The measurement process was essentially a mechanism where a set of metrics was proposed and data was collected to evaluate the values of these metrics. There are now frameworks for organizing measurement. Metrics are based upon models and high level characteristics and qualities are being defined in terms of an integrated set of metrics.

Refinement of Models and Measures

Metrics were assumed to be the same for all environments, i.e., it was assumed that the models for all environments were identical. We have begun to understand that a measurement program must be tailored to the needs of an organization and a customer and that different organizational and customer models offer different refinements of the models and metrics needed.

Automation of Measurement

Whatever automation existed was typically at the code level in terms of statement counts. Measurement systems are being developed that integrate measurement, and provide online feedback to managers and organizations.

Increased Application

Several companies were collecting measures but few were analyzing them and even fewer were using them in an effective way. This has changed. Many companies now have measurement programs with data collected across the organization.

This talk will discuss progress in all of the above areas.

References:


Bibliography:

The following is a set of papers that has appeared in the literature relevant to software measurement.


