On the Quality of Data

Thomas Ostrand

Accurate and reliable data is critical to derive conclusions from software engineering empirical studies. Our experience in collecting data from several large industry projects has taught us that raw data frequently is not what it seems, and that great care must be taken in interpreting information provided by software developers and testers.

A reliable contact person within a project, preferably someone who has been associated with the project for its entire lifetime, is an invaluable resource for describing the project's databases, and explaining how to access them. Such a contact can also provide insight into the project history, explain the project's culture, and frequently resolve the meaning of otherwise obscure entries in the databases.

Online software databases are typically populated both by programs and by human users. Both types of information can be misleading, as shown by the following examples:

- We collected data from 35 consecutive releases of a large system. The version control system automatically recorded the initialization date of each release, which is a key value for our models. It turned out that this recorded start date wasn't necessarily the real start date. Sometimes, the release was entered into the database well before it was actually populated with any files. We ended up using the first date of file population as the release start date.
- The problem report forms have a severity (1-4) associated with each MR. This value is supposed to indicate the importance of the problem, and be a guide to the urgency of fixing it. Unfortunately, we have found that the value chosen can sometimes be based on social reasons, rather than providing a realistic evaluation of the problem's importance.
- Fields in online report forms may have default values; if the user enters nothing, the default is used. If the project doesn't emphasize the importance of the user actually making a choice, this can seriously skew the result totals.

Recommendations for data collection:

- Understand exactly how fields in data collection forms are filled out
  - Are default values supplied if the user doesn't make a choice?
  - If possible, ask the project management to eliminate default values in forms.
  - What are the available options for users?
  - Do users choose from a drop-down list, from a radio list, or do they fill in free text.
- Understand the semantics of values that are provided
  - Fields such as category, phase, and severity may have different meanings to different projects, even within the same company.
  - The meanings of values may change over time even within a single project.
- Validate collected data
  - If the amount of collected data is manageable, examine each individual entry to assess its consistency and reasonableness.
  - If the data set is too large for individual examination, validate a randomly selected subset.

V. Basili et al. (Eds.): Empirical Software Engineering Issues, LNCS 4336, p. 102, 2007.