Measuring Requirements Traceability From Multiple Angles at Multiple Lifecycle Entry Points

Doron Becker, CISSP
Security and Privacy Professional Services, EDS
doron.becker@eds.com

Ideally, requirements traceability metrics are started at the beginning of a project. The initial statement of work or request for proposal form the root of the requirements tree and the traditional requirements metrics (requirements added, modified, deleted) are baselined from those documents. But what if you come in with the project already underway and, in all likelihood, behind schedule or over budget? In this presentation, the author shares his experiences in beginning the collecting of requirements metrics at various stages of the lifecycle – from architecture to testing to help desk support – based on work with a number of government and commercial clients.

One client had no requirements documentation at all but had analysis data in their problem report database. In this case, a module traceability matrix was constructed using data collected from the client's problem report database. This allowed the introduction of a metric showing how many subsystems were affected when a given module was changed. Since the number of hours needed to fix an individual module was known, the hours required to fix other subsystems could be estimated. A metric to collect the total number of hours required to close a problem report, based on the coupling metric, is being rolled out. The data made clear that the lack of well-crafted requirements added significant cost to the development effort and increased user frustration. The customer is now using the data to redesign and institutionalize the requirements collection and control processes.

A second client had a requirements database implemented via a well-known requirements tool, but little to no insight on requirements volatility and modification cost. For this client, a standard set of requirement traceability metrics, along with measures that look at average lifecycle time per requirements change and existing hours data, are helping to reduce their modification costs. This set of metrics included: number of requirements added, modified and cancelled during the given reporting period, along with the total number of requirements in effect at the end of the reporting period.

In September 2002, the author spoke at the eGOV Information Assurance Conference on requirements traceability and the Federal Enterprise Architecture Framework. Material from that presentation will be used to show how requirements traceability processes and metrics can give early insight into requirements volatility and implementation issues.

Similar to the costs of fixing defects, the challenges of starting requirements metrics collection substantially increase the later in the lifecycle one starts but the return on investment can still be significant at each stage.

Proceedings of the 11th IEEE International Requirements Engineering Conference
1090-705X/03 $17.00 © 2003 **IEEE**