“Blue Collar Formal Methods” in Commercial Quality Assurance

Richard Denney
Corporate Operations QA Manager
Landmark Graphics Corp.
Rdenney@lgc.com

There are many areas in commercial QA where Formal Methods (FM) could make useful contributions, yet there has not been a lot of focus by the FM community on commercial QA in a “real-world” setting. The use of FM by a software QA group in a commercial setting presents a different set of needs than might be encountered in the context of, say, FM as used by developers working on their own product in a commercial setting, or researchers working in FM [Denney et.al.] For example, QA groups often operate in a "MASH" like mode to support a much larger development group: there is no time for tidy, thorough operations on a single patient; rather surgeons are forced to triage and hastily perform rudimentary operations on a large number of incoming. For this type environment there needs to be increased focus on what I call “Blue Collar FM”, i.e. FM for the working folks. It’s about trying to identify that 20% of FM – as well as those 20% of applications of FM -- that gives developers & QA engineers 80% of the bang for the buck.

Example areas for applying Blue Collar FM are: test specification, project estimation metrics (e.g. function point like metrics based on formal specs), operational profiles and reliability engineering, and formal specification styles (e.g. algebraic and model-based) as a basis for natural language specification templates.

Another potentially fruitful area for the application of Blue Collar FM is in increasing the rigor of technical peer reviews. Formal technical peer reviews (e.g. walkthroughs & inspections) are popular in industry as a way to "test" a product early in the life-cycle before code is even written. Providing training on, and participating in, these peer reviews is “state of the practice” stuff for QA groups. The use of FM as an analysis tool of other peoples work has received some amount of discussion in the FM community, e.g. [Bowen and Hinchey]. Likewise in the QA community the potential leverage of formal methods in the technical review process has been recognized [Britcher], [Dyer 91], [Dyer 92], [Jackson and Hoffman], [Parnass and Weiss], [Van Emden].

But using FM to increase the rigor of analysis in the context of technical peer reviews adds some real world challenges:

- Reviewers are often times domain experts, rather than software engineers, with little or no experience with requirements specification, much less formal methods

- Reviewers are in some cases developers from other projects brought in to get an “outside view”, and hence are more or less looking at the documents for the first time; a practice encouraged by the technical peer review literature

- Preparation times by reviewers prior to the meeting is usually only 1 to 2 hours

Analysis under such constraints is a real challenge, but this is “state of the practice QA” and systematic analysis tools are needed for these types of reviews.

What would Blue Collar FM as applied to technical peer reviews look like? Perhaps the design of better inspection checklists that can be used by reviewers not trained in formal methods. If a reviewer doesn’t understand what a good specification is, how will they know a bad one when they see it? The structure of formal specification methods like the model-based and algebraic can contribute to knowing what information a good natural language specification should contain without necessarily delving into all the mathematics.

The author conducts a style of inspection that borrows from the depositions of [Votta] and the active design reviews of [Parnass and Weiss]. They are “depositions” in that the inspection minimally involves two people: the interviewer and the interviewee. They are “active” in that the interviewer, who must be versed in formal methods and has pre-reviewed the functional spec, walks the interviewee (usually a domain expert) through a light-weight modeling session of the functional specification (or some part thereof) under review as a way to generate questions that are then posed to the interviewee; this puts the interviewee in “active” review mode as opposed to “passive”. The advantages of this approach are:

- One gains independent (from the person who wrote the functional spec) review from a domain expert or experts
- One gains a more rigorous review with a minimal investment in formal methods training
- The cost efficiency of depositions (see [Votta] for details) offsets what some might complain is the added expense of using formal methods
- In-depth pre-meeting preparation time is incurred by the interviewer, then leveraged across the many interviews, with minimal preparation time by the interviewees.

As has been said [Kac], models aren’t so much to explain and predict, as to pose sharp questions, and that’s exactly the role that formal methods based modeling plays in this type of inspection.

In conclusion, I believe commercial QA groups as a consumer of FM is a fruitful yet largely unexplored area, with potential benefits for both QA and FM. For QA groups to leverage off FM however, a body of Blue Collar FM techniques needs to be identified which scales-up to the hectic MASH style operation of QA groups of the real world.

References

Bowen, Jonathan P. and Michael G. Hinchey, “Ten Commandments of Formal Methods”, Computer, Vol. 28, No. 4, April 1995

Britcher, Robert N., "Using Inspections to Investigate Program Correctness," Computer, Vol. 21, No. 11, Nov. 1988, pp. 38-44.

Denney, Richard, Dick Kemmerer, Mark Ardis (sitting in for Nancy Leveson), Alberto Savoia, Joint ISSTA / FMSP Panel: “What State-of-the-Art is not State-of-the-Practice”, Proceedings of the 1996 Int’l Symposium on Software Testing and Analysis (ISSTA), ACM Press, 1996

Dyer, Michael, "Verification Based Inspection," Proc. 25th Hawaii Int'l Conf. System Sciences, Vol. 2, IEEE CS Press, Los Alamitos, Calif., 1991, pp. 418-427.

Dyer, Michael, The Cleanroom Approach to Quality Software Development, John Wiley & Sons, New York, N.Y., 1992, pp. 96-99.

Jackson, Ann and Daniel Hoffman, "Inspecting Module Interface Specifications," Software Testing, Verification, & Reliability, Vol. 4, No. 2, June 1994, pp. 101-117.

Kac, Marc, “Some Mathematical Models in Science”, Science, 166, No 3906 695, 1969

Parnas, David L. and David M. Weiss, "Active Design Reviews: Principles and Practices," J. Systems and Software, No. 7, Elsevier, New York, N.Y., 1987, pp. 259-265.

Van Emden, Maarten H., "Structured Inspections of Code," Software Testing, Verification and Reliability, Vol. 2, No. 3, Sept. 1992, pp. 133-153.

Votta, Lawrence G. Jr., "Does Every Inspection Need a Meeting?," Proc. 1st ACM SIGSOFT Symp Foundations of Software Eng., ACM Press, New York, N.Y., 1993. Published in ACM Software Eng. Notes, Vol. 18, No. 5, Dec. 1993, pp. 107-114.