

Wise Computing

(abstract of invited lecture)

David Harel¹

The Weizmann Institute of Science, Rehovot, Israel

Major advances in languages, tools and methodologies have improved our ability to develop reactive systems, but the task remains difficult, expensive and error prone. One of the key reasons is the growing complexity of many kinds of reactive systems, which increasingly prevents the human mind from managing a comprehensive picture of all their relevant elements and behaviors. We present a vision that calls for a major change in the way complex software and systems are developed, by shifting the power balance between the human engineers and the development environment. In our computing paradigm, which we term *Wise Computing*, the development environment is turned into a much smarter, proactive, creative and interactive stakeholder in the development and maintenance processes. Ideally, the computer will join the development team as an equal partner—knowledgeable, concerned, and active.

The *wise development suite* (WDS) would interact with us wisely, like a colleague. It would respond to our needs with knowledge, and, utilizing extensive computing power “under its hood”, will proactively help in the variety of tasks that constitute the development process of the desired system. It should thus become a creative and proactive stakeholder, perhaps even a leader, in the development process. This will be manifested in it initiating discourse and actions based on deep insights into the system's structure and behavior, its overarching goals and rationale, and the environment in which it operates. It will use relevant knowledge (both general and domain-specific) to participate in the elicitation, formalization, validation and iterative enrichment of requirements, thus helping to increase confidence in the requirements, and establishing their consistency. And in the spirit of almost 30 years of model-driven development, the WDS will also be central to the ability to directly execute/simulate those requirements and/or translate them into running code.

The WDS will be able to explore, on its own, functionality and behavior both exhaustively and under various “what-if” conditions, communicating on multiple levels. Throughout development and maintenance, the computer will be

¹ This represents joint work with Guy Katz, Rami Marelly and Assaf Marron.

constantly investigating itself, in a sort of self-aware fashion. It will detect problems, including bad and conflicting behaviors, goals and requirements that are not met, inefficiency in execution, and unneeded complexities in specification and implementation. The WDS will then initiate and propose changes and enhancements,

The WDS vision calls also for runtime enhancements, where the system will be able to interact with users and with other systems in order to explain past behavior and allow the user to influence future behaviors. For example, a door in a chemical plant or an airplane will be able to explain to a human why it is presently closed, what will happen if it is manually opened, and discuss in detail sensor information and alternative sequences of manual and automated actions associated with opening and closing it.

The two-way interactions of the WDS will employ visual representations, examples, pseudo and conventional code. A key capability will be the use of natural language in both directions. Indeed, despite much work on natural languages for requirements and program specification, we are still far from the point where we can automatically read and parse requirements specified in a way that is natural and accessible to humans, and from them create a correct formal specification.

The most immediate benefit of a wise computing suite will be, of course, a significant reduction in the development time and cost of complex systems, and will result in much improved system quality. Run-time wisdom will increase user and regulator confidence in systems, further expanding development and adoption. And over and above all of this, we believe that in the farther future we will experience new dimensions of innovation, as rich new capabilities and new ranges of safety will be initiated (and often invented!) by wise systems, rather than only by humans.

How will the WDS do all this? Well, this is a vision, and for many facets thereof much research is required even to figure out they can be achieved. However, we have already done some work, and our ideas are explained in a paper we recently submitted for publication. A preliminary version can be found at: <http://arxiv.org/abs/1501.05924>. Also, we have built a modest and preliminary wise development suite, which we view as a promising proof-of-concept. The current version of the tool, as well as prerecorded video clips demonstrating its main principles on two examples, can be found at: <http://www.wisdom.weizmann.ac.il/~harel/CACM.wisecomputing>.

