

MobiCASE 2010
Workshop on Mobile Software Engineering
Position Paper

Paper title: **Bringing Software Engineering Practices to Mobile Application Development**

Name: Tony Wasserman

Role/Title: Professor

Organization: Carnegie Mellon Silicon Valley

Preferred Contact: tonyw@sv.cmu.edu

Please state your point of view on the issues of Mobile Software Engineering. Please specifically address the following 5 questions:

1. Your experience as a software engineer?

In the deep recesses of the 20th Century, I was responsible for the architectural design of several large-scale systems, most notably the commercial product Software through Pictures (StP). StP, the flagship product of Interactive Development Environments (IDE), was a multiuser software design environment that ran on a heterogeneous network of Unix workstations. It was one of the first commercial products to include open source software, one of the first to use the X-Windows system, and the first software development environment to use a relational DBMS as a shared repository. (StP is the foundation for the open source OpenAmeos tool, available from openameos.org.) I also developed the User Software Engineering methodology, which used an outside-in approach for developing interactive information systems, and featured tools for rapid prototyping of user interfaces. In addition to teaching and using numerous other software analysis and design methods, I even wrote a little code.

2. Your experience with mobile software development?

At Bluestone Software, I led an engineering team in the design and development of the Total-e-Mobile toolkit, aimed at helping developers of mobile applications for a commercial J2EE app server. We made these tools work successfully with several different mobile platforms of that era, shipping the first version of the product in 2000, and internationalizing the toolkit for several languages. Since then, I have regularly experimented with various platform and cross-platform mobile development tools and environments.

3. How does traditional software engineering relate to the engineering of mobile applications and systems?

Traditional software engineering provides the foundational principles for software development, independent of the application domain. These principles include modularization, separation of concerns, abstraction with architectural layering, use of a suitable development process, and attention to critical non-functional properties, including usability, robustness, scalability, performance, and efficient use of system resources, including processor, memory, and power utilization. Beyond that, there are well-established patterns, processes, and tools for the development of games and for Internet applications. Mobile application developers can (and should) draw on these practices and tools as they conceive and develop mobile applications.

4. What are the distinguishing features of mobile software specification, architecture, development and testing that need special attention, skills, or innovation?

Even though mobile software development draws heavily from traditional software development practices, there are some unique issues with mobile applications, most notably in testing. Testing of desktop and server applications is often complicated by hardware configurations, the presence (or absence) of various software components in the runtime environment, and by non-reproducible runtime effects in a multi-process networked environment. Real-time systems add the complications of timing constraints and the difficulty of handling sensor-based data. Mobile applications are not only affected by the same issues, but also by properties of the device.

5. What is the suggested focus and agenda for mobile software engineering research and education?

Mobile software engineering is a relatively new topic, and thus presents numerous opportunities for research and education. Topics related to the user experience are particularly relevant, and include not only interaction design, but also performance and speech recognition. At the same time, mobile software engineering research and education should not become isolated from the broader area of software engineering, since the areas of common interest far outweigh the unique areas. This topic is explored at greater length in the complementary paper, "Software Engineering Issues for Mobile Application Development," to be published in the proceedings of the Workshop on Future of Software Engineering Research, Santa Fe, NM, November, 2010.
