Innovations Syst Softw Eng (2013) 9:217
DOI 10.1007/s11334-013-0226-7

EDITORIAL

Software health management

Abhishek Dubey · Gabor Karsai ·
The Guest Editors of SI: SwHM

Received: 3 September 2013 / Accepted: 6 September 2013 / Published online: 15 September 2013
© Springer-Verlag London 2013

Software health management is defined as a technology that applies the principles and techniques of system health management to software systems. It is motivated by the apparent gap between the importance and complexity of software in today's cyber-physical systems and the rare, but undoubtedly present occurrence of software malfunctions in those systems. While engineers strive to create dependable systems unforeseen environmental conditions or faults in the hardware can trigger latent defects in the software with potentially negative consequences. The goal of software health management is to maintain system function and performance, even when software fails in unexpected ways.

System health management is a well-established discipline in aerospace systems: many air and space vehicles today have quite elaborate health management systems on board. Software fault tolerance techniques are also well-known and practiced since the days when computers were first used in critical applications. Software health management combines these directions: it borrows techniques like anomaly detection, fault diagnostics and mitigation from the first and techniques like triple modular redundancy and checkpoints and restarts from the second to manage the 'health' of the software system to maintain functionality and performance. Like in system health management, the goal of software health management is to prevent a software fault from becoming a system failure.

Software health management is not a formal discipline yet and its specific techniques are being worked out in a number of projects. This collection of papers aims at providing an overview of this new field. The paper by Srivastava and Schumann provides an introduction and justification why software health management is not only relevant but also necessary for safety critical systems where dependability is of the utmost importance. The rest of the collection follows the flow in a typical system health management system: anomaly detection, through fault diagnosis, to fault mitigation. The paper by Pike, Wegmann, Niller, and Goodloe shows how run-time monitoring and anomaly detection on software can be realized using a functional language. Person and Rungta introduce the technique of directed incremental symbolic execution that can be used to maintain the correctness of software monitors. The paper by Schumann, Mbaya, Menshoel, Pipatsrisawat, Srivastava, Choi, and Darwiche presents how Bayesian techniques can be used for anomaly detection and fault diagnostics for software systems. Finally, the paper by Mahadevan, Dubey, Balasubramanian, and Karsai shows how software fault mitigation can be implemented using a declarative specifications and general search algorithms.

This collection of papers represents the state-of-the-art for software health management. We hope that it will also motivate new research and development in this exciting new field.

A. Dubey (✉)
Research Scientist Institute for Software-Integrated Systems,
Vanderbilt University, Nashville, USA
e-mail: dabhishe@isis.vanderbilt.edu

G. Karsai
Professor of Electrical Engineering and Computer Science,
Vanderbilt University, Nashville, USA
e-mail: gabor@isis.vanderbilt.edu