

# Self-Healing Software

Goutam Kumar Saha

[sahagk@gmail.com](mailto:sahagk@gmail.com)

Self-healing systems represent a very new area of research that deals with fault tolerance for dynamic systems. Self-healing deals with imprecise specification, uncontrolled environment, and reconfiguration of systems according to their dynamics. The term “elf-healing” denotes the capability of a software system in dealing with bugs. Fault Tolerance for dependable computing is about providing the specified service through rigorous design whereas self-healing is about run-time issues. Software which is capable of detecting and reacting to its malfunctions, is called self-healing software. Such software system has the ability to examine its failures and to take appropriate corrections. Self-Healing system must have knowledge about its expected behavior in order to examine whether its actual behavior deviates from its expected behavior in relation of the environment. Self-Healing system in general has similar objectives to the area of dependable computing system. Techniques involved in self-healing are similar to those of dependable computing techniques but all dependable computing research areas are not self-healing. Self-healing categories of aspects include

- Fault-model or fault hypothesis
- System-response
- System-completeness
- Design-context.

A fault-model of Self-Healing system is to state what faults or injuries to be self-healed including fault duration, fault source such as, operational errors, defective system requirements or implementation errors etc.

System-response includes the aspects of fault detection, degree of degradation, fault response and an attempt to recovery action or compensation for a fault. Fault detection approaches involved in a self-healing system include application system’s semantics-driven assertions, supervisory checks, examining the computing answers, comparison of replicated components, online self testing etc. Complete restoration of functionality may not always be possible for a self-healing system after a fault. Such ability is limited by built-in redundancy in self-healing system. Techniques such as fault masking, retry, roll back or roll forward etc may be used for fault response.

The system-completeness aspect deals with reality of knowledge limits, incompleteness in specifications and designs thereof. It also deals with the problem of system self-knowledge, system evolution etc. Handling the architectural incompleteness for example, of third-party components or of various patches during or after system deployment is really a challenging issue in developing a self-healing system. Designers of Self-Healing application system should have a thorough knowledge about their application systems’ semantics. Designer’s inadequate knowledge about a system’s behavior in presence of faults is also a vital aspect in developing self-healing system. Field data is very useful to cope up this issue.

Design-context addresses the problems on abstraction level, component-level homogeneity, system linearity, system-scope, pre-deterministic behaviors, user involvement aspects etc. Self-healing system, a new paradigm of fault tolerance for dynamic systems, is yet to be matured.

Further Reading:

- D. Tosi, "Research Perspectives in Self-Healing Systems," Report of the University of Milano-Bieocca, 2004.
- P. Koopman, "Elements of the Self-Healing System Problem Space," Proceedings of the ICSE WAD03, 2003.
- R. d. Lemos, "ICSE 2003 WADS Panel: Fault Tolerance and Self-Healing," Proceedings of the ICSE 2003.
  
- Goutam Kumar Saha, "Application Semantic Driven Assertions toward Fault Tolerant Computing," ACM Ubiquity, Vol. 7, No. 22, pp. 1 - 27, ACM Press, 2006, USA.
- Goutam Kumar Saha, "Software Fault Tolerance through Run – Time Fault Detection," ACM Ubiquity, Vol. 6, No. 46, pp. 1-5, ACM Press, 2005, USA.
- Goutam Kumar Saha, "Software Based Fault Tolerance – a Survey," ACM Ubiquity, Vol. 7, No. 25, pp. 1-17, ACM Press, 2006, USA
- Goutam K Saha, "Software Based Fault Tolerant Array," IEEE Potentials, Vol. 25, No. 1, pp. 41-45, IEEE Press, Jan-Feb 2006, USA.
- Goutam Kumar Saha, "Transient Fault Tolerance through Algorithms," IEEE Potentials, Vol. 25, No. 5, pp. 25-30, IEEE Press, Sep-Oct 2006, USA.
- Goutam Kumar Saha, "Fault Tolerance in Web Services," ACM Ubiquity, Vol. 7, No. 9, pp.1-8, ACM Press, 2006, USA.
- Goutam Kumar Saha, "Software Based Transient Fault Tolerance," International Journal of Mathematics and Computer Science, Vol. 1, No. 2, pp. 179-190, 2006, France.

Author's Biography:

In his last nineteen years' research & development and teaching experience, he has worked as a scientist in LRDE, Defense Research & Development Organization, Bangalore and at the Electronics Research & Development Centre of India, Calcutta. At present, he is with the Centre for Development of Advanced Computing, Kolkata, India, as a Scientist-F. He is a fellow in IETE and senior member in IEEE, Computer Society of India, and ACM etc. He has received various awards, scholarships and grants from national and international organizations. He is a referee of CSI Journal, AMSE Journal (France), IJCPOL (USA), IJCIS (Canada) and of an IEEE Journal / Magazine (USA). He is an associate editor of the ACM Ubiquity (USA) and of the International Journal of Computing and Information Sciences (Canada). His field of interest includes software based fault tolerance, web technology and Natural Language Processing.

Source: Ubiquity Volume 8, Issue 12 (March 27 2007 - April 2, 2007)  
<http://www.acm.org/ubiquity/>

