A

Automatic Self Healing Using Immune Systems

Junaid Ahsenali Chaudhry
Ajou University, South Korea

Seungkyu Park
Ajou University, South Korea

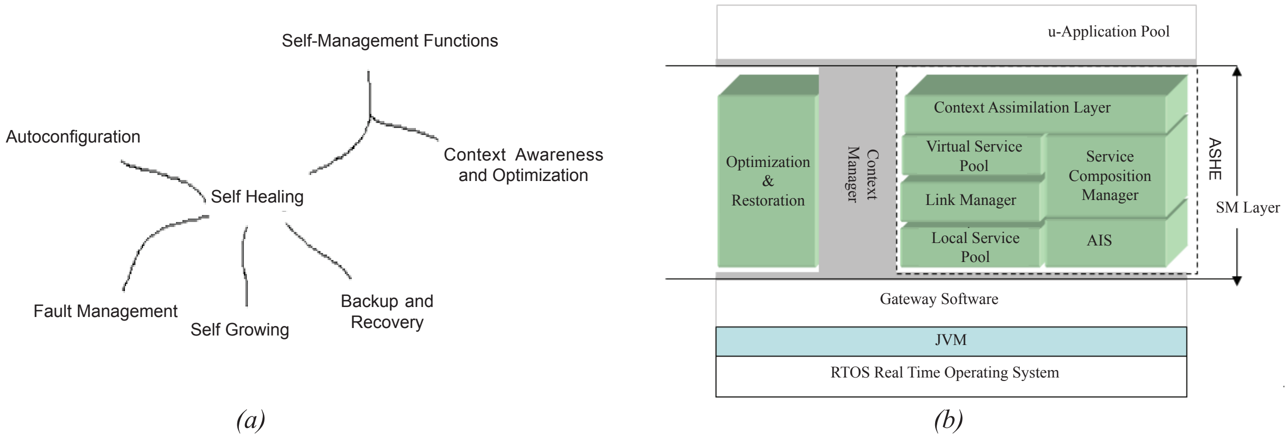
INTRODUCTION

The networking technologies are moving very fast in pursuit of optimum performance, which has triggered the importance of non-conventional computing methods. In the modern world of pervasive business systems, time is money. The more the system fulfills the needs of the requesting user, the more revenue the business will generate. The modern world is service-oriented, and therefore, providing customers with reliable and fast service delivery is of paramount importance. In this article we present a scheme to increase the reliability of business systems. The arrival of ubiquitous computing has triggered the need previously mentioned even further, and people hold high exceptions from this technology. In Morikawa (2004), the authors characterize the vision of ubiquitous computing into two categories: "3C everywhere and physical interaction." 3C consists of "computing everywhere," "content everywhere," and "connectivity everywhere." "Physical interaction" connects the hidden world of ubiquitous sensors with the real world. This wide area of coverage and high scalability makes a ubiquitous system quite fragile toward not only external threats, but internal malfunctioning too. With the high probability of "abnormal behavior" it is more important to have knowledge of fault and its root causes. As described in Yau, Wang, and Karim (2002), application failures are like diseases, and there can be many types of faults with matching symptoms, thus fault localization and categorization are very important. Unlike in Hung et al. (2005) and Steglich and Arbanowski (2004), we cannot categorize all abnormal functionalities into fault tolerance or (re)configuration domains simply because faults do not have any predefined pattern; rather we have to find those pattern. Moreover, as in Steglich and Arbanowski (2004) the "without foresight" type of repair in ubiquitous systems is desired. The conventional FCAPS (Fault, Configuration, Accounting, Performance, Security), network management model categorizes management functions in one group, but we argue that categorizing management functions into different segment is mandatory in self management paradigms.

Since in highly dynamic and always available very wide area networks, one fault can be atomic (caused because of one atomic reason) or it can be a set of many faults (caused because of many atomic or related reasons). It is often a good practice to break the problem into smaller atomic problems and then solve them (Chaudhry, Park, & Hong, 2006). If we classify all different types of faults (atomic, related, and composite) into one fault management category, the results would not be satisfactory, nor would the system be able to recover from the "abnormal state" well. Since the side effects of system stability and self healing actions are not yet known (Yau et al., 2002), we cannot afford to assume that running self management modules along with functional modules of the core system will not have a negative effect on the system performance. For example, if the system is working properly, there is no need for fault management modules to be active. Lastly, instead of having a fault-centric approach, we should have a recovery-centric approach because of our objective that is to increase the system availability

In this article we present autonomic self healing engine (ASHE) architecture for ubiquitous smart systems. We identify the problem context through artificial immune system techniques and vaccinate (deploy solution to) the system through dynamically composed applications. The services involved in the service composition process may or may not be related, but when they are composed into an application they behave in a way it is specified in their composition scheme. The vaccines are dissolved to liberate the system resources (because they take the system's own resources to recover it) after the system recovery. When the system is running in a normal state, all self management modules are turned off except context awareness and self optimization.

Figure 1. (a) Proposed classification, (b) ASHE: Layered architecture



These two are always on to monitor and optimize the system respectively.

BACKGROUND

In this section we will compare our work with RoSES project, and Semantic Anomaly Detection Research Project.

Robust self-configuring embedded systems (RoSES) (Ricci & Omicini, 2003) is a project that explores graceful degradation as a means to achieve dependable systems. It concentrates on allocating software components to a distributed embedded control hardware infrastructure, and is concerned with systems that are entirely within the designer's control. Our system is a quasi-distributed linear feedback system that uses loose coupling of components (except with SCM). Every time a component fails, it is replaced by a slightly different component. This approach gives a powerful configuration control during fault diagnosis and removal process. Reconfiguration through replacing the failed component is one aspect of our system. The scalability of this system is confined to a local area network. With the increasing size of network the scalability is reduced drastically. Our target area is not local area networks, rather wireless ad hoc networks, and especially ubiquitous networks. The heterogeneity levels at large MANET scale may cause the system to lose its self-configuring value, as the increasing type of devices will make the scenario very complex.

The Semantic Anomaly Detection Research Project (Shelton, Koopman, & Nace, 2003) seeks to use online techniques to enter specifications from under specification components (e.g., Internet data feeds, etc.) and trigger an alarm when anomalous behavior is observed. An emphasis of the research is using a template-based approach to make it feasible for ordinary users to provide human guidance to the automated system to improve effectiveness. The template-based systems are not robust. We need to have some hybrid approach in order to manage the different devices and real time systems. Moreover, our system uses online anomaly detection through IIS detectors, but it also uses the goods of dynamic service composition for more accurate problem addressing. The following is the architecture of the system proposed.

SYSTEM MODEL

Figure 1(a) describes the functional organization of SH and SM functions. Context awareness (CA) and self optimization (SO) are the functions that are constantly in action. The auto configurations (AC), fault management (FM), and so forth are classified as on-demand sub functions in ASHE architecture. Frequently a system faces complex problems that cannot be solved by merely reconfiguring the system. So to ensure the system restoration, we place healing as the super set. Figure 1(b) shows the block architecture of ASHE at a residential gateway level.

Related Content