

TECHNICAL NOTE

Available Online at www.jgrcs.info

AUTONOMIC COMPUTING: A LONG TERM VISION IN COMPUTING

Sandeep Kumar Chauhan* and Dr Arun Sharma

*(Research Scholar Mewar University Chittorgarh, Rajasthan)

(Professor and Head, Dept. of Comp. Sc. and Engg., Krishna Inst. of Engg. and Tech., Ghaziabad)

Abstract: Autonomic computing is the latest trend in computing which aims building more scalable and reliable systems. The main objective of Autonomic system is to provide ease of use and no maintenance to the end user. In this paper we will discuss the autonomic computing approach and its future challenges.

INTRODUCTION

Autonomic computing is generally considered to be a term first used by IBM in 2001 [1] to describe computing systems that are said to be self-managing. IBM has defined four basic properties for autonomic systems. These properties are **Self-CHOP** (self- configuration, self-healing, self optimization and self-protection).

However the concept of self-management and adaptation in computing systems has been around for some time. A high penetration of IT system has resulted into complex and heterogeneous system which are very difficult to manage. As a result, applications, programming environments and information infrastructures are rapidly becoming brittle, unmanageable and insecure. The Autonomic computing aims to provide a zero cost maintenance and highly reliable system to end user at the same time providing a transparent interface to the complex infrastructures [2].

MAJOR PROPERTIES OF AUTONOMIC COMPUTING

IBM has defined four basic properties for autonomic systems [4] . These properties are:

- a. **Self-configuration:** an autonomic computing system configures itself according to high-level goals, i.e. by specifying what is desired, not necessarily how to accomplish it. This can mean being able to install itself based on the needs of a given platform and the user.
- b. **Self-optimization:** an autonomic computing system optimises its use of resources. It may decide to initiate a change to the system proactively (as opposed to reactive behaviour) in an attempt to improve performance.
- c. **Self-healing:** an autonomic computing system detects and diagnoses problems. If possible, it should attempt to fix the problem, for example by switching to a redundant component or by downloading and installing software updates. However, it is important that as a result of the healing process the system is not further harmed, for example by the introduction of new bugs or the loss of vital system settings. Fault-tolerance is an important aspect of self-healing. Typically, an autonomic system is said to be reactive to failures or early signs of a possible failure.
- d. **Self-protection:** an autonomic system protects itself from malicious attacks but also from end users who

inadvertently make software changes, e.g. by deleting an important file. The system autonomously tunes itself to achieve security, privacy and data protection. Thus, security is an important aspect of self-protection.

AUTONOMIC COMPUTING APPROACH

An autonomic system should be self-contained, self aware and adaptive to the external environment. It should able to start-up , operate and recover from error scenarios without any manual intervention or external help. The approach of autonomic computing is to migrate the complete knowledge into the system. This will help to reduce the manual intervention with the system. The system will keep learning from previous experiences and will use the past experience to solve the future problems. Autonomic computing is about shifting the burden of managing systems from people to technologies. For Autonomic Computing to reach these goals will require open standards and technologies.

The fundamental approach of an autonomic computing system is control loop [4] that accesses and controls the managed resource in cyclic manner.

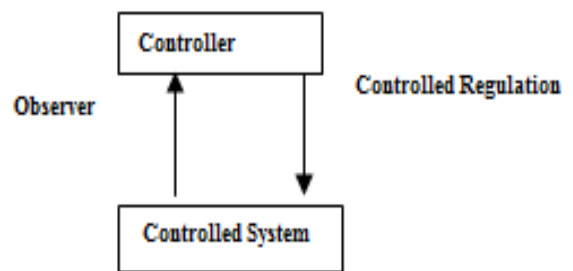


Figure 1: Generic Control loop

The controller can control the controlled system via feed-forward control strategy or feed-back control strategy. With a feed-back control strategy, regulation is determined based on the comparison of the difference between a desired objective and measured output delivered by the controlled system.

This helps the system always up and running. According to IBM, hundreds or even thousands of these control loops are expected to work in a large-scale self-managing computer system.

DIFFERENT LEVEL IN AUTONOMIC COMPUTING

Autonomic computing is an evolutionary process and consists of five levels [5]. These levels are:

Basic:

The basic level represents the starting point where a significant number of IT systems are today. Each element of the system is managed independently by systems administrators who set it up, monitor it, and enhance it as needed.

Managed:

At the managed level, systems management technologies are used to collect information from disparate systems into one, consolidated view, reducing the time it takes for the administrator to collect and synthesize information.

Predictive:

At the predictive level, new technologies are introduced that provide correlation among several elements of the system. The system itself can begin to recognize patterns, predict the optimal configuration and provide advice on what course of action the administrator should take. As these technologies improve, people will become more comfortable with the advice and predictive power of the system.

Adaptive:

The adaptive level is reached when systems can not only provide advice on actions, but can automatically take the right actions based on the information that is available to them on what is happening in the system.

Autonomic:

Finally, the full autonomic level would be attained when the system operation is governed by business policies and objectives. Users interact with the system to monitor the business processes, and/or alter the objectives.

BENEFITS AND FUTURE APPLICATION SCOPE OF AUTONOMIC APPLICATIONS

There are short term and long term benefits of autonomic systems. The short term benefits are in terms of cost saving as autonomic systems are easy to scale up. The idle processing can be use utilised fully through networked systems. The systems will be highly available, stable as there will be lesser error as systems will be self-healing. The long term benefits will be to achieve end to end service management. Another long term benefit will be to embed autonomic capabilities in client or access devices, servers, storage systems, middle-ware and network itself to make them more robust.

Autonomic computing can be implemented in various areas like electricity, railway management, air traffic control systems etc. Few of the possible application areas are discussed as follows:

Electricity Transmission Systems:

Current electricity transmission systems employ manual management of interconnected subsystems that supply electricity to the users. This manual management is inefficient and error prone. To remove these shortcomings, autonomic capability can be integrated to these systems. The sensor could monitor the environment factors such as

temperature, humidity, wind speed, month, etc and autonomously provide the necessary power. Any short-circuit could also be detected and as a measure to correct it supply is halted in that node immediately and a worker is notified to for error correction. This would greatly reduce the probability of power-cuts and dynamically provide electricity as required. Self-optimization can be thought of in terms of providing load forecasting facility on the basis of previous electricity requirements. System will also provide security against threats such as unauthorized accesses.

Train Route Management System:

Introduction of high-speed trains has posed the need for a faster management of the whole system. Manual management is quite slow as compared to management through computers. Also the level of imprecision in these real time systems is highly dynamic. Train schedules are also dynamic. Autonomic computing can provide this speed of decision making. This technique of using technology to manage technology has real benefits to these real time systems. The system will configure itself upon introduction of new trains and new tracks. This will also decrease collision probability or any slowdown due to failure at any point. System will minimize cost and time of service providing efficient service in a convenient way. This system will too provide security against threats such as unauthorized accesses.

Multiprocessor System:

Both versions of multiprocessing i.e. symmetric as well as asymmetric are hard-coded and have their advantages and shortcomings also. The processors can be configured to process their specific processes. But this account for performance slowdown when there is load on only one processor. This provides the motivation to find a way to manage the processor load autonomously. Self-management capability is the key to successful multiprocessing where each processor is running their own instance of operating system. With autonomic capability the throughput, resource utilization, performance of the system will increase.

Traffic Control System:

Centralized architecture of current traffic control systems will soon become brittle and unmanageable as the traffic volume and number of devices used as a sensor to provide situation of traffic at a time increases. Thus, there is a need of a decentralized system that consists of autonomic devices capable of monitoring the environment and communicates with other adjacent autonomic devices so as to make decisions for the complete traffic management. Self-configuration nature can be built as the capability to identify new node and take it into working use. If a node fails others around must notify and take charge of its working. This system will also provide users with current traffic condition so that they can reach their destination conveniently.

Travel Guide:

A Travel-Guide application can be developed which is capable to provide location-dependent service in response to a change in the user's location. GPS system can be used to track a user and provide this input to an autonomic element which can then search for available services in that locality in its database.

ISSUES AND CHALLENGES IN AUTONOMIC COMPUTING

Autonomic computing system has many issues [3]; these are related to design & development, testing, implementing and measuring the reliability of autonomic system.

The autonomic computing design approach is a current research topic. There are various issues related to design approaches. Design of autonomic systems is a complex task. The designing of autonomic system can be based on Hard Computing and Soft Computing. Hard computing is based on mathematical approaches to problem solving and it imbibes their basic characteristics from mathematics. While, soft computing methods are based on biological approaches to problem solving, where mathematics does not play central role as it does in engineering problem solving methods. Thus, soft computing provide ability to handle imprecision, uncertainty & partial truth that is inherently present in real world systems. Hard computing provides stability with high precision solution.

Soft Computing comprises of Artificial Intelligence, Artificial Neural Networks, Genetic Algorithms, probabilistic reasoning, Genetic Programming, chaos theory and sections of learning theory [6].

Current systems lie in domain of hard computing, but now Soft Computing principles are being applied to provide tolerance for imprecision, uncertainty and partial truth that is inherently present in any complex real world problem. The autonomic concepts of self awareness, environment awareness, self adjustment and autonomous behaviour all reside within Artificial Intelligence (AI) of which Soft Computing has emerged as a collection of methodologies that are particularly tolerant of imprecision, uncertainty and partial truth.

Testing autonomic elements [7] and verifying that they behave correctly will be particularly challenging in large-scale systems because it will be harder to anticipate their environment, especially when it extends across multiple administrative domains or enterprises.

Testing networked applications that require coordinated interactions among several autonomic elements will be even more difficult. It will be very difficult to build Test Systems that capture the size and complexity of realistic systems and workloads. It might be possible to test newly deployed autonomic elements by having them perform alongside more established and trusted elements with similar functionality. The measurement of reliability of autonomic systems is also a major challenge. It is still an area of research to calculate the reliability of the autonomic systems.

Another issue in Autonomic Computing is the measurement of automaticity [8] of the IT complex system. Another open research area is to correctly calculate the automaticity degree of Autonomic Computing Systems.

CONCLUSION

Autonomic computing is an emerging holistic approach to computer system development that aims to bring a new level of automation to IT systems. We believe autonomic computing will lead to breakthrough in the way. Computing research has evolved in many isolated, loosely coupled research fields as (security, fault tolerance, high performance, AI, network, agent systems, effective event management etc.). Each discipline has managed to deliver computing systems and services that meet their target domains (e.g, High Performance Computing, Fault Tolerance Computing [9]). However, if the computing systems and application would require combining these capabilities; i.e., delivering computing systems that provide high performance, secure, fault-tolerance, intelligent computing systems and applications are not practically feasible and available. Autonomic computing is the emerging computing field that addresses all these issues in an integral way and can be viewed as the computing field that will converge all these disciplines in one field – Autonomic Computing. However, there are still many challenges that must be addressed by the research community for this new computing paradigm.

REFERENCES

- [1]. IBM, "Autonomic Computing: IBM's Perspective on the State of Information Technology"; <http://www.research.ibm.com/autonomic/>
- [2]. J. Kephart and D. Chess, "The Vision of Autonomic Computing," IEEE Computer 36(1): 41-50 (2003).
- [3]. Dr Arun Sharma, Sandeep Chauhan and Dr P.S Grover, "Autonomic computing : Paradigm Shift for Software Development" , CSI India ISSN 0970-647X Vol 35 Issue No 5 Aug 2011
- [4]. IBM, "An Architectural Blueprint for autonomic computing" , V7, June 2005
- [5]. Mohammad Reza Nami, Mohsen Sharifi "A Survey of Autonomic Computing Systems" IFIP International Federation for Information Processing, 2007, Volume 228, 2007
- [6]. R. Sterritt, B. Smyth and M. Bradley: PACT- Personal Autonomic Computing Tools.
- [7]. Myers, Glenford J. (1979). The Art of Software Testing. John Wiley and Sons. ISBN 0-471-04328-1.
- [8]. Schneider, W. & R. M. Shiffrin. (1977). Controlled and automatic human information processing: 1. Detection, search, and attention. Psychological Review, 84, pp1-66
- [9]. R. Sterritt, "Facing fault management as it is, aiming for what you would like it to be", Soft-Ware: 1st International onference on Computing in an Imperfect World, Belfast 8-10 Apr., LNCS2311, Eds. D.W. Bustard, W. Liu, R. Sterritt, Springer-Verlag, 2002