Available online www.jocpr.com

Journal of Chemical and Pharmaceutical Research, 2014, 6(2):187-192

Research Article

ISSN : 0975-7384
CODEN(USA) : JCPRC5

A new universal architecture of resources management information system for the national park based on the autonomic computing

Zhiyun Feng, Maozhu Jin* and Renpei Yu

Business school, Sichuan University, Chengdu, China

ABSTRACT

Resources management information system is designed for national parks based on the autonomic computing, which can reduce complexity of the large system. For researching, four concepts were defined at first, then a new architecture was designed and also given an example, the next, an autonomous behavior generator and an autonomous element were designed based on the idea of robot autonomy control and expert system, and new conceptual levels were also listed. It is a new universal architecture not only used in the national park but also implemented in other fields with similar necessary and problems.

Key words: resources management; information system; autonomic computing; autonomic element; autonomous behavior generator

---

INTRODUCTION

Nowadays, many information systems are widely implemented in the distinctive national park of China, or some regions, even province coverage. Those systems are relevant to many managerial fields and applied some advanced technologies, such as GIS/GPS, digital monitoring, RFID and so on. Those kinds of new technologies can improve the management performance but there are some difficulties restricted the application.

Firstly, most of systems lack the unified infrastructures and open standards. Information is not shared with each other, more than that, those are hardly unified in one system so as to hardly get accurate and complete information without delay. Secondly, it is a terrible waste of repeatedly designing and developing similar systems. So are repeatedly arranging same devices and machines for different systems in one place. Thirdly, most of those architectures are centralized management, hardly to improve and repair. Sometimes, a little fault could breakdown the whole system. Last but not least, most national parks are in remote areas, lack of human resources, so the system has to be easy to used and repaired.

Hence we try to design a new universal architecture of the resources management information system with the idea of autonomic computing. It is possessed of stronger applicability and generality, improved service quality with intelligence, can be applied with most kinds of managerial functions in parks, regions, provinces, even nation coverage.

A REVIEW OF AUTONOMIC COMPUTING

Autonomic computing is an emerging research hot-topic, which aims at reducing management complexity for human users by means of “technologies managing technologies”. Paul Horn advanced the concept from neural science in 2001[1] [2]. The autonomic nervous system is widely distributed in visceral organs with mutual connection of the neural network. In the autonomous nervous system, “brain” looks like a "manager", which need not "know" how to operate each organ (“equipment”) but only need to design the plan and direct behaviors. So is autonomous computing system. In 2003, Kephart systematically discussed the vision of the autonomous computing, and pointed out the essence is self-management, including four aspects: self-configuration, self-optimization, self-repairing and self-protection [3]. Tianfield added self-planning, self-learning, self-scheduling and self-evolution etc. [4]. Sterritt added the self-control, self-adaptive, self-recovery and self-diagnosis [5]. In addition, Ganek, White, Tesauro, Parashar and De Wolf are further expounded the concept in more specific technical background. Therefore, although the concept is still in the development process, the basic meaning is more and more clearly [6-10].

At present, the field of the autonomous computing has made significant progress, typically including: Kephartproposed MAPE (monitor - analysis - planning - execution) control ring as the self-adaptive control mechanism of the autonomic element and introduced a unified framework that interrelates three different types of policies: Action, Goal, and Utility [11][12]. After that, Kephart pointed out that it requires a combination of a variety of science and technology, including: system science, software engineering, management policy, modeling and some artificial intelligence. Kephart also took a utility function policy reflecting managers' goals [13] [14]. Parashar presented AutoMate, a conceptual model and the system structure of autonomous computing, against complex, dynamic, isomerism and uncertainty network environment. AutoMate includes three main parts: the Accord programming system [15] [16] [17].Pen proposed a method based on AOSE (service oriented software engineering) to support different levels of policy [18]. Hinchey and Sterritt put forward the concept of self-management software, which is expected to put automation, autonomy and reliability to a new level [19] [20].

Above all, autonomous computing is developing rapidly in theory and technology. But from the whole point of view, the research is still in its infancy [21] [22].But even so, the united and practical architecture of resources management system could accomplish with the help of the idea of autonomous computing.

DEFINE SOME CONCEPTS

Definition 1: Resources management information system (RMIS)

Resources management information system (RMIS) is managed all resources that are owned or used or monitored or managed by a park or an organization, such as flora and fauna, platform, weather, vehicles, equipment, finance, human resources and so on. Moreover, it is a universal architecture with AEs. RMIS can be expanded not only in a national park, but a region, even the whole country, so that the macro and micro management can be strengthened with low cost at the same time . RMIS is a software system, whatever choosing any advice or equipment, such as, RFID, GIS, GPS, access machinery, the architecture need not be changed.

Definition 2: Autonomous Element (AE)

Autonomous element (AE) is the basic structure block of system, which consists of one autonomic behavior generator (ABG) and some managed resources. AE manages internal resources by the ABG and be managed through outward standardized interface. Every AE has its own mission, and completes task in its range. If out of range, other AEs or human managers will intervene or replace it. In a word, AEs comprise the system.ABG achieves the management tasks through the control cycle of MAPE, "monitor - analyze - plan - execute". According to the current state, ABG can take many kinds of behaviors depend on managerial policy. AEs are inconsistent, with sequence, periodicity, intermittently, pulse, delay and burst. From the perspective of control theory, many AEs have time variability, nonlinear and uncertainty [23][24]. That means AE need accept and process all kinds of data. Of course, AE has also duplicability which looks like the component technology, but with intelligence. So it can be reproduce, upgrade and maintenance quickly and easily.

AE is expressed as triads:

$$AE = (ABG, MR, interface) \quad (1)$$

The ABG provides management instructions to managed resources, and the MR is managed resources (or other managed AE).

Definition 3: Autonomous Behavior Generator (ABG)

The autonomous behavior generator (ABG) is defined as: the center of the autonomic element (AE). All policies are sent form the ABG. Under the guidance of the policies, ABG guides AE to realize self-planning, self-learning etc.

Definition 4: Policy Based Management

The policy in the paper is further defined as: policy is the managerial rules to guide the behavior of the autonomic element. Generally, policies are divided into two classes: obligation type and authorization type, which respectively means "should/should not do" and "allow/not allowed to do".

DESIGN AN UNIVERSAL ARCHITECTURE OF RMIS

Based on the autonomous computing, the universal architecture of RMIS can be built as a large system composed with several levels and some parts, includes: system, sub-system, module, sub-module, till AE, and the relationships in the same layer are parallel. Each AE has three basic capabilities: data processing and perception, analysis and decision-making, planning and implementation. AE is an intelligent block without human operating, but RMIS does not reject the human intervening. In a word, AE manages the normal and human does the abnormal. To explain the architecture, shown in table 1, the system is simply divided only seven parts and three levels.

Table.1The Architecture of RMIS

System	RMIS						
Sub-System	1.Bio-physical resources management sub-system	2. Tourist management sub-system	3. Transportation management sub-system	4. Operation sub-system	5. Marketing & Service sub-system	6. Safety sub-system	7. Emergency sub-system
Modules	1.1 Get information 1.2 Process information 1.3 Send information 1.4 Decision	2.1 Access control 2.2 Intelligent guide 2.3 Broadcast 2.4 Multimedia	3.1 Line settings 3.2 Vehicle scheduling 3.3 Line planning 3.4 Decision	4.1 Human Resources 4.2 Finance 4.3 Materials 4.4 Equipment	5.1 E-commerce 5.2 Virtual Tour 5.3 Ads 5.4 After-service	6.1 Vehicle Safety 6.2 Flora & fauna Safety 6.3 Tourists Safety 6.4 Platform	7.1 Emergency Monitoring 7.2 Emergency Rapid Response 7.3 Police Dispatch 7.4 Tourists Dredging

To facilitate research, assumed only three levels, and the module is the AE. Shown in fig. 1, each block represents an AE, each number of AE corresponds to the number of table 1.

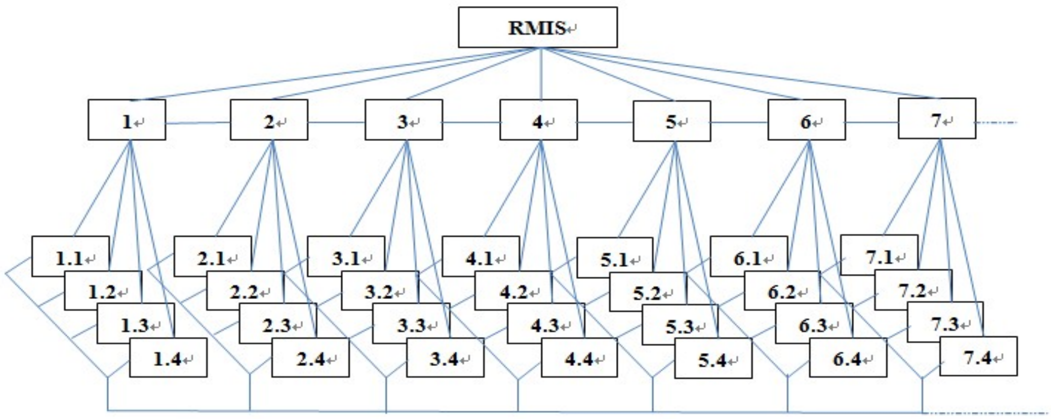

Fig. 1 The Model of RMIS based on AE

The RMIS shown in Fig. 1 can be likened to the management team, which is composed of one “general manager” and seven “department managers”. Every “department manager” has own “permission” and “capability” to manage some AEs in some extent. The “permission” and “capacity” are provided in the overall design. All AEs can run in one computer or distributed in some computers based on the wired or wireless network. Obviously, if we use the available technology to build it, it will be too complicated to realize. But based on intelligent AE, RMIS can easily complete it. First, every AE has its own mission, and can accomplish all normal tasks which can be pre-programmed. Those AE can not only reproduce at developing stage, but also be easily used, maintenance and upgrades as an independent unit. Secondly, AE is an autonomous unit, and some AE can manage other AEs. Only when the AE does not solve the problem, a human manager need intervene. So, the architecture is clearly divided into concentration, autonomy and scattered concentration. This is a simple architecture, easy to implement.

BUILD AN ABG

For saving developing time and cost, AE should have some versatility and interchangeability. AE generates autonomous behaviors which need to be divided into different levels according with structural environment and non-structural environment. The capacity of ABG is divided into 6 levels (from 0 to 5), to form six concepts inductively. It can be increased and decreased as necessary.

Table.2 The capability levels of ABG

Levels	Characteristics	Explanation	Example
0		Decision analysis is not required, and ABG is not settled.	5.3, 2.3, 2.4
1	Pre-programmed	Prepare the plans in advance according to the experience, and store them into the knowledge base, then select the appropriate plan to do according to the field data.	1.1, 1.3, 2.1
2	Self-re-planning	Add real-time re-planning on the basis of pre-programmed.	3.1, 3.3, 5.2
3	Self-re-computing	Add re-computing on the basis of self-re-planning to realize optimization, evolution, and adaptive.	1.2, 3.2, 4.1, 4.2, 4.3, 4.4, 7.1
4	Self-learning	Add learning on line and update knowledge base on the basis of self-re-computing.	1.4, 2.2, 3.4, 7.2, 7.3
5	Self-diagnosis	Auto-diagnose with pre-program.	Every ABG

CONSTRUCT AN AE

AE is designed based on the theory of autonomy control of robot. The essence of autonomy control has three aspects: perception, decision-making and execution. In this architecture, AE(shown in fig.2) is composed of seven blocks: ABG is a center, and others are managed resources, including perception, planning, upload, issue, joint with other AEs, and interact with other AEs.

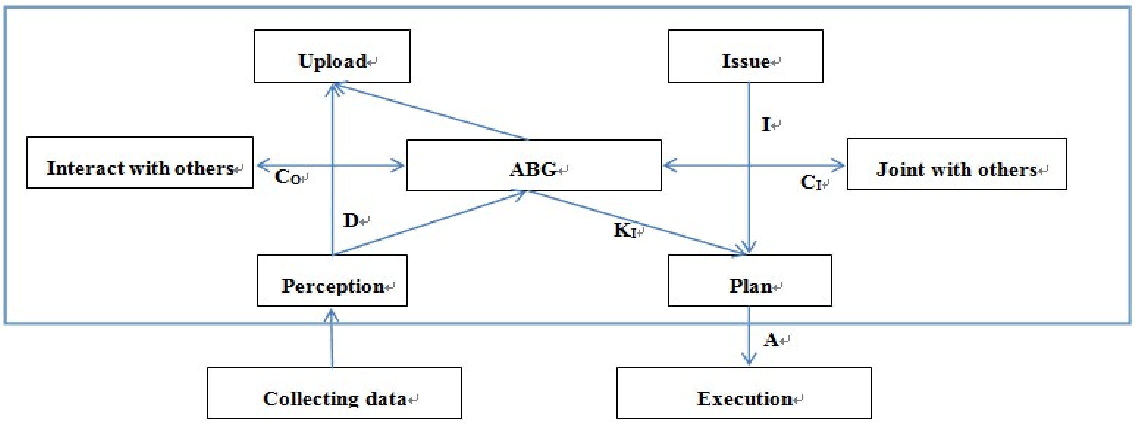

Fig. 2The Model of a typical AE

The "actions" from ABG should be broadly interpreted as the results of the analysis and decision-making. Generally, five-tuple（D, C_o, C_i, K_i, I）can be used to describe AE. That is,

$$f_d:\ D \times C_o \times C_i \rightarrow K_i \quad (2)$$

$$f_p:\ K_i \times I \rightarrow A \quad (3)$$

Where D is the collection of inputting data. C_o is the collection of instructions to other AEs at the same layer. I is the collection of instructions from the upper layer. C_i is the collection of instructions from other AE. K_i is the collection of autonomous instructions by itself. A is the collection of outputting instructions. f_d is the mapping from inputting collection to behavior collection. f_p is the mapping from action to last inputting.

Autonomous behaviors can be realized in many of modern artificial intelligence methods, such as: expert systems, agent, contextual reasoning, knowledge systems, etc. Though the function of each module is different, the basic framework of data and interface are changeless between the parts. For example, in fig. 2, the element of diverting tourists (7.4) can be used the theory of the expert system to construct an ABG. There is a schematic of the common expert system (shown in fig. 3).

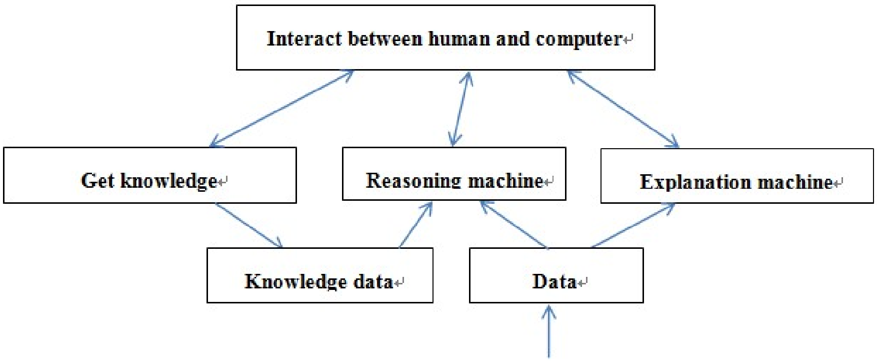

Fig. 3A model of ABG with Expert system

To illustrate the problem, there is another simplified model as shown in fig. 4.

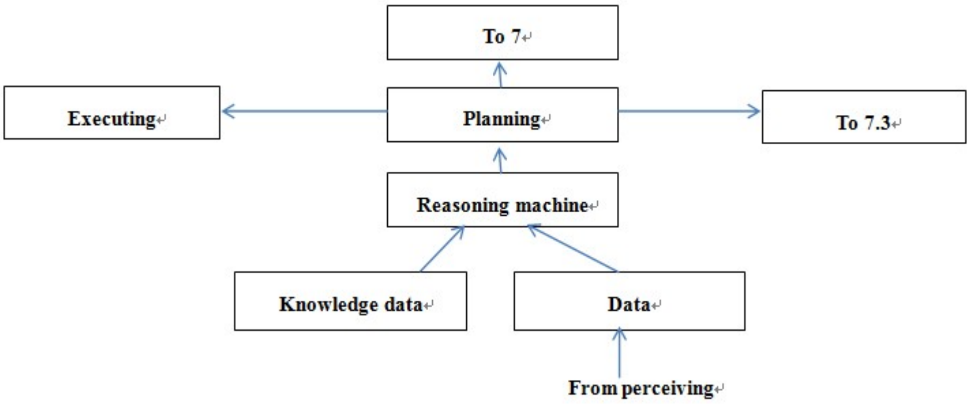

Fig. 4 A Model of ABG with Expert system (2)

At first, there are all kinds of pre-planning for emergencies in the knowledge data. When the emergent information gets to the data, reasoning machine will start. Then pre-planning will run with comparing, reasoning and decision-making, between reasoning machine and knowledge data. Then instructions from the planning module are sent to the field execution module, such as, diverting people and vehicle, start emergency signs. It also issued request to 7.1 and 7.3, such as: starting emergency monitoring, mobilizing of police, reporting to the superior in element 7. At last, element 7 will use similar process to generate some actions in sequences on a larger scale, such as some control and blockade.

CONCLUSION

The architecture of RMIS is an effective structure to solving the complex problems of a large system, and it is also a universal structure not only used in the national park but also implemented in other fields with similar necessary and problems. Comparing with the current methods of software development, to realize the architecture, still need re-engineer the processes of system and re-configure the resources of system.

Acknowledgments
The authors wish to thank the Major International Joint Research Program of the National Natural Science Foundation of China (Grant No. 71020107027), the National Natural Science Foundation of China (Grant No. 71001075) and Humanities and Social Sciences project of The Ministry of education of China (Grant No.12YJC630023), under which the present work was possible.

REFERENCES

[1] Horn P. *Autonomic Computing: IBM's Perspective on the State of Information Technology*, http://www.research.ibm.com/autonomic/manifesto/autonomic_computing.pdf, **2001**.
[2] Bear MF, Connors BW, Michael A, Wang JJ. *Neuroscience: Exploring the Brain*, Beijing: Higher Education Press, pp. 472−496 (in Chinese), **2004**.
[3] Kephart JO, Chess DM. *IEEE Computer*, Vol. 36(1), pp. 41−50, **2003**.
[4] Tianfield H. *Artificial Intelligence Review*, pp.57-91, **2005**.
[5] Sterritt R, Parashar M, Tianfield H, Unland R. *Advanced Engineering Informatics*, Vol 19(3), pp.181−187,**2005**.
[6] Ganek AG, Corbi TA. *IBM System Journal*, Vol42 (1), pp. 5−18, **2003**.
[7] White SR, Hanson JE, Whalley I, Chess DM, Kephart JO. *An Architectural Approach to Autonomic Computing*. In: Sunderam V, Das R, eds. Proc. of the ICAC 2004. Washington: IEEE Computer Society Press,pp. 2−9, **2004**.
[8] Tesauro G, Chess DM, Walsh WE, Das R, Segal A, Whalley I, Kephart JO, White SR. *A Multi-agent Systems Approach to Autonomic Computing*. In: Jennings NR, et al., eds. Proc. of the 3rd Int'l Joint Conf. on Autonomous Agents and Multiagent Systems. New York: ACM Press, pp. 464−471, **2004**..
[9] Parashar M, Hariri S. *Autonomic Computing: An Overview*. In: Banâtre JP, et al., eds. Proc. of the Int'l Workshop on Unconventional Programming Paradigms (UPP 2004). Berlin: Springer-Verlag, pp.247-259, **2005**.
[10] De Wolf T, Holvoet T. *A Taxonomy for Self-properties in Decentralised Autonomic Computing.* In: Parashar M, Hariri S. eds. Autonomic Computing: Concepts, Infrastructure, and Applications. Boca Raton: CRC Press, pp. 101-120, **2007**.
[11] Kephart JO, Chess DM. *IEEE Computer*, Vol 36(1), pp. 41-50, **2003**.
[12] Kephart JO, Walsh WE. *An Artificial Intelligence Perspective on Autonomic Computing Policies*. In: Verma D, Devarakonda M, Lupu E, Kohli M, eds. Proc. of the 5th IEEE Int'l Workshop on Policies for Distributed Systems and Networks. New York: IEEE Computer Society, pp.3-12, **2004**.
[13] Kephart JO. *Research Challenges of Autonomic Computing*. In: Griswold W, et al., eds. Proc. of the 27th Int'l Conf. on Software Engineering. New York: ACM Press, pp. 15-22, **2005**.
[14] Kephart JO, Das R. *IEEE Internet Computing*, Vol. 11(1), pp.40−47, **2007**.
[15] Parashar M, Li Z, Liu H, Matossian V, Schmidt C. *Enabling Autonomic Grid Applications: Requirements, models and Infrastructures*. In: Babaoglu O, et al., eds. Proc. of the Int'l Workshop on Self-Properties in Complex Information Systems. Berlin: Springer-Verlag, pp.273−290, **2005**.
[16] Parashar M, Liu H, Li Z, Matossian V, Schmidt C, Zhang G, Hariri S. *Cluster Comput*, Vol.9(2), pp.161−174, **2006**,.
[17] Liu H, Bhat V, Parashar M, Klasky S. *An Autonomic Service Architecture for Self-managing Grid Applications*. In: Katz DS, ed. Proc. of the 6th IEEE/ACM Int'l Workshop on Grid Computing. Seattle: IEEE Computer Society Press, pp.132−139, **2005**.
[18] Peña J, Hinchey MG, Sterritt R. *Towards Modeling, Specifying and Deploying Policies in Autonomous and Autonomic Systems Using an AOSE Methodology*. In: Sterritt R, et al., eds. Proc. of the 3rd IEEE Int'l Workshop on Engineering of Autonomic & Autonomous Systems (EASE 2006). Washington: IEEE Computer Society, pp.37-46, **2006**.
[19] Hinchey MG, Sterritt R. *IEEE Computer*, Vol. 39(2), pp.107−109, **2006**.
[20] Franke C, Theilmann W, Zhang Y, Sterritt R. *Towards the Autonomic Business Grid*. In: Sterritt R, et al., eds. Proc. of the 4th IEEE Int'l Workshop on Engineering of Autonomic and Autonomous Systems (EASe 2007). Washington: IEEE Computer Society, pp.107−112, **2007**.
[21] Farha R, Leon-Garcia A. *Operations Research Methods for Autonomic Resource Management*, Journal of Network and Systems Management, **2009**.
[22] Ruth P, Rhee J, Xu D, et al., *Autonomic Live Adaptation of Virtual Networked Environments in a MultidomainI nfrastructure*, Journal of Internet Services and Applications, pp.141-154, **2011**,.
[23] Sterritt R, Hinchey M. *Adaptive Reflex Autonomicity for Real-time Systems,* Innovations in Systems and Software Engineering, pp.107-115, **2009**.
[24] Mello R. *Journal of Ambient Intelligence and Humanized Computing*, pp.11-33, **2011**.