http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 7 (5): 771-775, 2007
ISSN 1812-5638

Case-Based Reasoning Computation Migration Mechanism

[1,2]Wang Weihong, [1,2]Cao Yuhui and [1]Qin Zheng
[1]School of Electronics and Information Engineering Xi'an Jiaotong University, Xi'an 710049, China
[2]Hebei University of Economics and Business, Shijiazhuang 050061, China

Abstract: Aiming at the contradiction question between terminal resources limited and terminal computation in mobile collaboration, a case-based reasoning computation migration mechanism (CBRCMM) was proposed. At first, the key factors influencing computation migration were abstracted from mobile collaboration environment and they were taken as main characters of the computation migration case. Secondly, adopt the value interval and grading methods to depict the main characters, Thus, which were then used to enhance ability of computation migration case. Thirdly, apply appointment priority and easy adaptation priority principle and select the appropriate case relative to the current situation by the matching degrees of migration case. At last, modify the case based on the specific application practice in current situation; make it as a new case and save it in succeed record or fail record. At present, the mechanism has been applied to the Mobile Collaboration Platform of Supporting Mobile Collaboration Services. The practice indicates mechanism that this simple and effective can alleviate the contradictory question between terminal resources limited and terminal computation in mobile collaboration in a certain extent and support powerful mobile collaboration.

Key words: Mobile computing, mobile collaboration, computation migration, case-based reasoning

INTRODUCTION

The work of this study is inscribed in project which aims to constructing a platform of supporting mobile collaboration services to provide mobile user the ability of information access and collaboration with each other.

Along with the continuous development of mobile communications and the increasing popularity of mobile terminal devices, people can access any network resources and services through a variety of means at any time and any place (Satyanarayanan, 1996). However, because of the information increasing socialization, people gradually recognize that mobile computing should not only provide the capacity of accessing network resources and services to people, but also provide the capacity of mutual collaboration to people at any time, anywhere. It is said that people need collaboration by closer to the natural way without restrictions of time and space (Luff *et al*., 1998). Therefore, the mobile collaboration (Litiu and Zeitoun, 2004; Yushun *et al*., 2003) which integrated mobility and cooperativity has become a hot research field.

The inherent characteristics of mobile computing enabled mobile collaboration face more challenges than traditional computer-supported collaborative work together, such as limited resources terminals, the wireless link bandwidth fluctuations, unreliable transmission and so on. It has attracted much attention on how to resolve these issues.

Computation migration provide a efficacious method for the these issues. Many people have made a lot of work in the field, such as Wong *et al.* (1997) presented an infrastructure for collaborating mobile agents, named Concordia, Chenggang and Zhongzhi (2002) gave a module-based mobile Agent and its schedule method and Simon and Berthold (2005) provided an adaptive routing with stale information and so on. Acharya *et al.* (1997) is firstly cognizant of the influence to MA migration path, which made by hardware resource and its use state.

Brewington *et al.* (1999) provided the notion of TAP, Traveling Agent Problem, to program migration path for MA. But the previous work didn't consider taking full advantage of the existed knowledge in migration history. Although Simon and Berthold (2005) have considered the advantage the stale information, in the process of computation migration, the restriction of collaboration group in mobile collaboration environment was not taken into account. So, the Case-Based Reasoning Computation Migration Mechanism, named CBRCMM, was proposed in this study.

Corresponding Author: Wang Weihong, School of Electronics and Information Engineering Xi'an Jiaotong University, Xi'an 710049, China

Case-based reasoning: Case-based Reasoning (CBR) (Aamodt and Plaza, 1994) is an important new emerging reasoning technique in artificial intelligence field recently. It is different to rules-based reasoning pattern, such as it can obtain the solution for current problems through accessing the similar case in case base (Gavin and Zhaohao, 2003). The general process of it is as follow.

Firstly, when a new problem was met, the system search most similar candidate case to the problem being to be settled from the primal case base based on the key features and reused the case.

Secondly, if the solution of the candidate case is not satisfied, it can be modified to suit the problems and the final revised case was saved as a new case in the case base.

In addition, it is usually divided into solution problems type and interpretation type (David *et al*., 2002).

The cases are the knowledge cells in CBR, reasoning processes are similar to the human experience analogy reasoning. The knowledge acquisition and express is natural and directly. And it has itself learning ability. So, it has successfully applied in many areas, such as JUDGE, CABOT, CASSIOPEE etc. Therefore, this research adopted case-based reasoning methodology to achieve the computing migration in mobile collaboration platform.

CBRCMM MODEL

Computing migration is a complex dynamic process in Mobile collaboration computing environment because of not only considering migration occasion but also considering the whole effect of collaboration group after migration.

Because of the diversification of mobile cooperative terminals and the difference of function config, the relationship among mobile cooperative node became complex very much. Such as, the mainly mobile terminal in mobile cooperative computing include intelligence mobile phone, PDA (Personal Digital Assistant), mobile personal computer and so on, their running speed and storage capability have great difference. In order to conquering the performance difference of mobile terminal to ensure the group collaboration, the computation tasks on the nodes with weaker computation ability can be migrated to the nodes with better computation ability under some specified situation. When and How the computation migration should be done is a key issue. If we can adequately use the existed experience in previous cases, the above issue will become facility.

Now, we discuss a supposition, there are nine collaborators in a collaborative group and each collaborator takes a mobile terminal. Thus, the situation of terminal devices and their config is shown in Table 1.

In Table 1, there are three kinds of mobile cooperative terminal devices with difference function config, which are mobile PC, PDA and intelligence mobile phone, respectively. There are six kinds of attributes in Table 1 and 1 means the terminal devices supports the relevant function and 0 are reverse. The symbol ---- means indeterminacy.

Generally, the parameters about each kind of terminal have also some different, but the difference of parameters about different kind of terminal are great. The parameters about average performance of Mobile PC, PDA, Intelligence Mobile Phone used in our project is as follow Table 2, respectively.

From Table 2, we can know clearly their running speed and storage capability have great difference.

The main factors impacting computing migration: Let r_i is the factor impacting computing migration. Then, the set of all factors impacting computing migration was defined as R, R is as follow.

$$R = \{ri, I = 1,2,3,4....\}$$

In general, the impact of each factor is different for computing migration. Thus, let the weight of each factor in computing migration decide is ω_i, the weight vector is defined as follow.

$$W = (\omega_1,\omega_2,...,\omega_i,...)$$

Table 1: List of terminal devices and their config

	Item						
Col. ID	Terminal type	Intant message	Image browse	Mobile phone voice	Computer audio	Computer video	Other config
1	Mobile PC	1	1	0	1	1	-
2	PDA	1	1	0	0	0	-
3	Intelligence Mobile Phone	1	1	1	0	0	-
4	Intelligence Mobile Phone	1	1	1	0	0	-
5	PDA	1	1	0	0	0	-
6	PDA	1	1	0	0	0	-
7	Intelligence Mobile Phone	1	1	1	0	0	-
8	Mobile PC	1	1	0	1	1	-
9	Intelligence Mobile Phone	1	1	1	0	0	-

Table 2: The parameters list of average performance of mobile PC, PDA, intelligence mobile phone

	Parameter	
Terminal	Main work frequency	Cache capability
Mobile PC	2.0G	512M
PDA	300M	64M
Intelligence mobile phone	200M	16K

Assume there are N factors, the sum of their weight is 1, it is said

$$\sum_{i=1}^{N} \omega_i = 1$$

Here, based on the practice we choose computing resource in mobile terminal device, power consume of battery, network bandwidth in group collaboration work states and correspond communication response time during completing collaboration work task as the main factors impacting computing migration. Namely r_1 is computing resource, r_2 is power consume of battery, r_3 is network bandwidth and r_4 is communication response time. Then, $R = (r_1,r_2,r_3,r_4)$, namely, R is the set of computing resource, power consume of battery, network bandwidth and communication response time.

The main factors impacting computing migration acted as the main features of the migration case in this paper.

Case storage: Two important issues were considered in case storage, namely, the storage mechanism of case and express of case.

The data structure used for case storage is varied. The methods relatively ease to use, such as cases were recorded as a relationship. In which, one part of data recorded the main features being to be used to matching. Another stored the corresponding solution step of the issue aiming at corresponding main features. There are the methods relatively complex to use, such as cases were expressed by proof-trees. There are also other commonly methods to be used, such as case expressed as a very large rule set of the condition-action, the fact used to describe condition is the most outstanding features in case; the operation composed action is the solution used in new condition. The case storage mechanism relatively easy was adopted in this study.

Case expression not only determines the conversion from real world to the case, but also has great impact for the efficiency of case reasoning. A qualified case expression should include at least two parts: explaining information about issue (the initial conditions); achieving the goals of the solution about issue. To enhance the generalization ability of case, using interval replace point value approach and grading. For example, point value of four factors in migration situation, energy consumption, network bandwidth, communications corresponding time, the computing resources, possible can not happen again in a fairly long period of time and it is just like a fleeting moment. To capture its value, its value may have been changed. Therefore, the point value is difficult to operation and application. Therefore, the main features will be stored the value expressed as an interval in case, like $[a_{r_i}, b_{r_i}]$. In order to depict the dynamic and uncertain of environment, the concept of grade level fuzzy membership was used in its grading levels. The method is as follow.

Step 1: Set the neighbour of each factor as Δ_{r_i}

Step 2: Grading levels to $[a_{r_i}, b_{r_i}]$ based on Δ_{r_i} and sort it according to order.

Step 3: The point value of main feature r_i of case at a time is v_{it}, the fuzzy membership of it vest in the level is μ_{ik}.

For example the interval of memory resource is $v_{it} \in [1,64]$, $\Delta_{r_i} = 4$, then the grade table as Table 3. The data in table was normalization for easy narration.

If $v_{it} = 7$, the membership of it vest in level 2 is 100%. The membership of other level it belonging to was calculated based on similarity to each level.

The rest may be deduced by analogy.

Case matching: In order to reuse the previous knowledge, we must identify the main features and construct a mapping. The mapping defined how to make the previous experience apply to current situation. It is said how to mapping problem space to solution space.

Def.2 Case set: Let C be the set of n cases, $C = \{c_n: n \in N\}$. c_n is a case.

c_n includes two parts: The part of main features storage, the part of migration scheme storage. It is as follow.

$$c_n = (G_{r_1}, G_{r_2}, G_{r_3}, G_{r_4}; D, A, R, \cdots)$$

Table 3: Grading level of every factor

	Factor			
Level	Memory resource	Computing resource	Network resource	Communication response time
1	1-10	1-5	100-200	0.1-0.2
2	11-20	6-10	201-300	0.3-0.4
-	-	-	-	-
n	70-80	11-15	700-800	0.8-0.9
-	-	-	-	-

In above formula, G_{r_1} express the membership level of r_1, G_{r_2} express the membership level of r_2, G_{r_3} express the membership level of r_3, G_{r_4} express the membership level of r_4. D, A and R are the part of migration scheme storage. The D express migration decision-making, A express migration goal, R express migration path.

Def.2 Main features matching: Let main features were captured by the context awareness module in application system is, respectively g_1,g_2,g_3 and g_4. Their match degree respective with G_{r_1},G_{r_2},G_{r_3} and G_{r_4} are measured by their adding weight broad Euclidean distance. The Euclidean distance is defined as follow.

$$d = \sqrt{\sum[(\omega_i \mid G_{r_i} - g_{r_i} \mid]^2}$$

Therefore, choose the case satisfied min(d)<ε based on the principle of appoint priority. ε is matching threshold.

Modify and store case: Modify case indicates evaluation and revision for solution scheme. Generally, there are two ways: structural modifications and induced changes (Karjoth *et al.*, 1997). And the revision process was divided into two steps: Firstly, analyze the different between the new requirements and the candidate cases; Secondly, modify candidate case. The revised case after certification, if it is feasible or correct, the case can be stored as a new case to the case base. Thus, with the accession of new cases, it marks the system have completed a learning process.

CBRCMM APPLIED IN SMCSP

Application of CBRCMM: The SMCSP is a mobile electronic commerce Platform Supporting Mobile Collaboration Service, which was built over hybrid networks including wire network and wireless network with MAS technology and FIPA theory. It not only can provide the capability of accessing network resources and services, but also can support the mobile collaboration work among mobile users in anytime at anywhere. For implementing the cooperation among mobile users well, the CBRCMM was used in the SMCSP. The function module of it is shown as follow Fig. 1. The computation migration decision process includes four sub-modules mainly in Fig. 1, such as Capturing Context module, Feature Extraction for Cases module, Case-Based Reasoning module and Computation Migration Decision module.

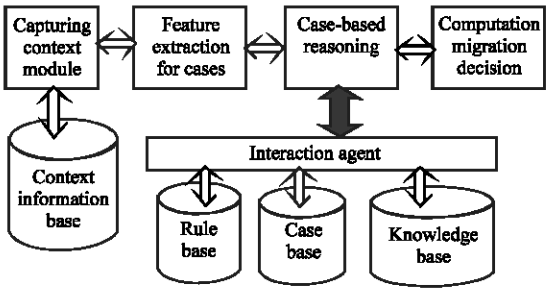

Fig. 1: The computation migration module in SMCSP

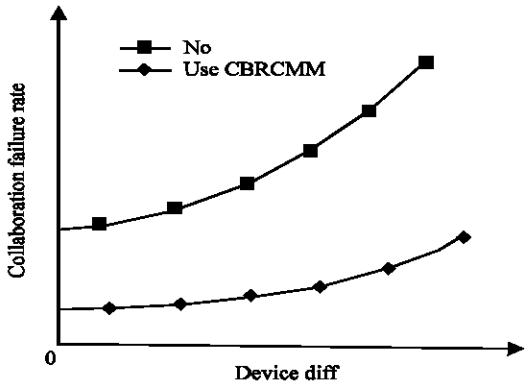

Fig. 2: Compare using CBRCMM and not using under different devices

Above that, the function of Capturing Context module is that capture the context information about migration case and save them in context information base; the function of Feature Extraction for Cases module is that extract the feature information of case form context information; the function of Case-Based Reasoning module is that carry on case-based reasoning by dint of rule base, case base and knowledge base. The rule base, case base and knowledge base have formed the slave subsystem, which implemented the support for the case-based reasoning by interaction Agent. The function of Computation Migration Decision module is that make decision based on the results of case-based reasoning.

In Capturing Context Module, the main context include computing resource context, power consume of battery context, network bandwidth context and communication response time text. This module interact with the module of Context Information Base and Feature Extraction for Cases, it store the context information in Context Information Base and provide data for the module --Feature Extraction for Cases.

The function of feature extraction is mapping the context information captured to a feature space for matching the existed cased. And take the value in a time point into a range value.

Analysis of CBRCMM Application: In SMCSP, while the list of mobile terminal devices and their configures are as Table 1 and the parameters list of average performance of mobile terminal devices are as Table 2, based on four factors: Memory Resource, Computing Resource, Network Resource and Communication Response Time, the experiment results are as Fig. 2.

From Fig. 2, we can see the use of CBRCMM reduces the collaboration failure rate, improve the performance of SMCSP.

CONCLUSIONS

As a new and important research embranchment of mobile computing, mobile cooperation can provide user more function supporting especially cooperative work capability. The computation migration mechanism gave a strong and effectual method to ensure the implementation of mobile collaboration. In this paper, we proposed a model CBRCMM and apply it to SMCSP. Because of taking full advantage of the preexisted knowledge in previous migration history, CBRCMM can enhance the efficiency of computation migration. And it considered the restriction of collaboration group, so it fit to the mobile collaboration environment. At present, the mechanism has applied the Mobile Collaboration Platform of Supporting Mobile Collaboration Services. The practice indicates the mechanism is simple and effective and it can alleviate the contradictory question between terminal resources limited and terminal computation in mobile collaboration in a certain extent and it can support powerful mobile collaboration. The practice shows the model is effectual and facile actualization. Our future research work is: To improve the model; and to farther validate the models.

ACKNOWLEDGMENT

This research is supported in part by the National Grand Fundamental Research 973 Program of China (Grant No. 2004CB719401), the National High Technology Research and Development Program of China (863 Program, No. 2003AA412020).

REFERENCES

Aamodt, A. and E. Plaza, 1994. Case-based reasoning: Roundational issues, methodological variations and system approaches. Artificial Intelligence Communications, IOS Press, 7: 39-59.

Acharya, A., M. Ranganathan and J. Saltzl, 1997. Sumatra: A language for resource-aware mobile programs. Proceeding of Mobile Object Systems: Towards the Programmable Internet, Berlin: Springer, pp: 111-130.

Brewington, B., R. Gray and K. Moizumi, 1999. Mobile Agents in Distributed Information Retrieval. Intelligence Information Agent, Berlin, Springer Verlag, pp: 355-395.

Chenggang, W. and Sh. Zhongzhi, 2002. Module-based mobile agent and its schedule method. J. Software, 13: 1628-1636 (Chinese).

David, W., N.R. Patterson and G. Mykola, 2002. Efficient similarity determination and case construction techniques for case-based reasoning. ECCBR., pp: 292-305.

Gavin, F. and S. Zhaohao, 2003. R5 model for Case-Based Reasoning. Knowledge-Based Systems, 16: 59-65.

Karjoth, G., D.B. Lange and M. Oshima, 1997. A security model for aglets. IEEE Internet Computing, 2: 68-77.

Litiu, R. and A. Zeitoun, 2004. Infrastructure support for mobile collaboration. Proceedings of the 37th Annual International Conference on System Sciences, Hawaii, pp: 31-40.

Luff, P. and C. Heath, 1998. Mobility in Collaboration. Proceedings of ACM Conference on Computer Supported Cooperative Work, ACM Press. New York, pp: 305-314.

Satyanarayanan, M., 1996. Mobile Information Access. IEEE Personal Communications, 3: 26-33.

Simon, F. and V. Berthold, 2005. Adaptive routing with stale information. Proceedings of the 24th Annual ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing, pp: 276-283.

Wong, D., N. Paciorek and T. Walsh *et al*., 1997. Concordia: An infrastructure for collaborating mobile agents. Proceedings of the Mobile Agents 1st International Workshop. Berlin: Springer-Verlag, LNCS., 1219: 86-97.

Yushun, L. and Sh. Meilin, 2003. Mobile collaboration writing. PIMRC2003, 1: 849-853.