

Research on Multi-Agent system for embedded mobile devices business applications

Haizhen Wang^{1, a *}, Zuozheng Lian^{2, b}

¹School of computer and control engineering, Qiqihar 161006, China

²Computer center, Qiqihar 161006, China

^awanghaizhen1976@163.com, ^blianzuozheng@163.com

Keywords: Embedded mobile device, Business applications, AUML specification, Muti-Agent system

Abstract. The mobility and intelligence of Agent make it have a very good application value, moreover, the study and application schemes on Agent model for embedded mobile devices are rare, so the article proposes an Agent model for the demand of embedded mobile devices business applications, which is consist of wireless mobile compute environment model, business intelligence Agent general framework, Agent core memory model, they is designed as a multi-system combining Agent technology with Belief Desire Intention model and memory evolution thought, and AUML specification is used to draw Agent design diagram during implementing system, finally server and client operating environment is set, and multi-Agent system for embedded mobile devices business applications is performed. The system functions are simple and easy to use.

Introduction

Along with the rapid development of network information technology, electronic commerce has been widely applied to the traditional business areas, people can use it to achieve online business transactions, in the transaction process an important activity is the negotiation [1], with what consultative mechanisms to improve the efficiency and significance of electronic commerce transactions plays an important role. In the 1990s, advanced Agent technology have been proposed, its mobility and intelligence make it have a very good value, and has widely used in mobile compute, distributed information retrieval, network manufacture, electronic commerce fields. Traditional electronic commerce by Agent technology has been realized on the PC, however, with the increased popularity of embedded mobile devices and mobile commerce applications, how to use Agent technology for embedded mobile devices business applications has become research focus [2].

Analysis of the current Agent technology applications research results for embedded mobile devices, Agent model and its application program are rare, and Agent models proposed by domestic universities and scientific research institutions have full function, but very complicated, not easy to achieve in low-and-middle embedded mobile devices whose calculating ability are limited. Multi-Agent system for embedded mobile devices business applications is implemented in the article, system function is streamlined, convenient to use.

Analysis of system function

The article firstly analyzes Agent model, and then realizes Multi-Agent System for embedded mobile devices business applications. Because Agent model is the basis of Agent autonomy and adaptability, the information processing characteristics of action rules, knowledge storage, learning and decision-making mechanism included in the model is the core of Agent, Agent information processing methods can imitate thinking of man, and memory is an important aspect. Therefore, the core function of Multi-Agent System is how to design a model which is suitable for embedded mobile devices, and on basis of the model implement better auto business negotiation. Using Agent feature, combining BDI Agent [3] structural features with memory evolution [4,5,6] information processing characteristics, a novel Agent model for embedded mobile devices is performed in the article, namely wireless mobile compute model based on Agent memory-evolution is designed, and based on the model Agent

auto-negotiation algorithm is designed, finally the model is performed implements by establishing server operating environment on the PC, designing client Agent program for embedded mobile devices.

Detail design of system

The section introduces the design of wireless mobile compute model which is consist of wireless mobile compute environment model, business intelligence Agent general framework and Agent core memory model, and uses AUML to specify Agent design drawn in order to achieve the system.

Wireless mobile compute environment model are consist of servers and clients provides the hardware and software environment for business negotiations, servers includes client registration server, data server, commerce medium server, application server, etc., clients includes PDA, Smartphone, notebook, and pc. Registration server is mainly responsible for registering information and providing services; Data server is accountable to store data corresponding service, such as the seller's products information; commerce medium server is mainly used as Agent negotiation and trading places; application server is a server with other Agent service functions, it can register to the registration server, and add the system to form a commerce Agent alliance.

Tree systems is designed for the model, namely the seller Agent system, the buyer Agent system and registration system, the relationship between them is shown in Fig. 1. The embedded mobile Agent of the buyer Agent system can carry buyer needs moving to the server, and interacts with the address lookup Agent, and address management Agent of registration system to get the seller service management Agent address, and passed to the commodity inquiry compare Agent, so that it can interact with the service management Agent to filter consultation commodity, and sends the information to the buyer negotiation transactions Agent to negotiate transaction, Finally, the consultation results are delivered to the buyer and seller embedded mobile Agent and return.

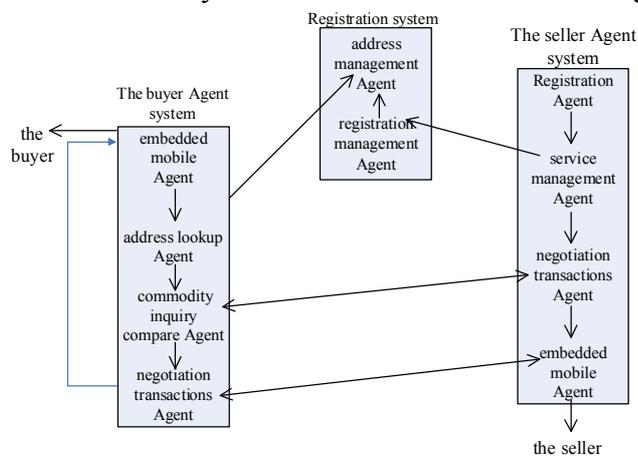


Fig.1. Multi-Agent system in wireless mobile compute environment

The intelligence Agent general framework is consisted of Agent core engine, knowledge base, belief vector, target base, sensor, communicator, scheduler, the function module interface, extension plug-ins. The main function of each part is as follows: Belief vector mainly stores Agent current belief information, also is an Agent short-term memory component. Knowledge base is mainly stores all the relevant information about Agent belief, also is an Agent long-term memory component, knowledge base exist in database form in the server, and XML file format in the embedded mobile client. Goal base mainly stores Agent relevant action object. Sensor receives feedback information from external environment. The information is passed to the Agent core engine after processing to carry out actions assessment and provide the basis for Agent decision-making. Communicator is an important part achieving negotiation, which realizes communication between Agents. Scheduler mainly realizes scheduling action and programming goal. Function module interface mainly stores and defines Agent conduct the action. Extension plug-ins is mainly used to achieve extensions. Depending on Agent role can extend different functions, such as inference rules, negotiation strategy, and coordination mechanism. Agent core engine is the core component which realizes decision-making and learning.

The system needs to adapt to the characteristics of embedded devices, so Agent super class is designed, and then design various Agents by inheriting super class. Each Agent is required to design appropriate function in accordance with their respective tasks in multi-Agent system. Agent super class designed by AUML is shown in Fig. 2. Agent super class has the following attributes: AID represents Agent identification, inherited from JADE Agent. AgentState represents Agent survival state which is consist of creation, activity, suspending, waiting, deletion, movement, etc. Role Agent represents Agent roles, such as buyer or seller. Goal represents Agent target. Agent super class has the following methods: setup (), destroy (), takedown (), move (), execute (), wait (), wakeup (), resume (), suspend (), etc. inherited from JADE Agent.

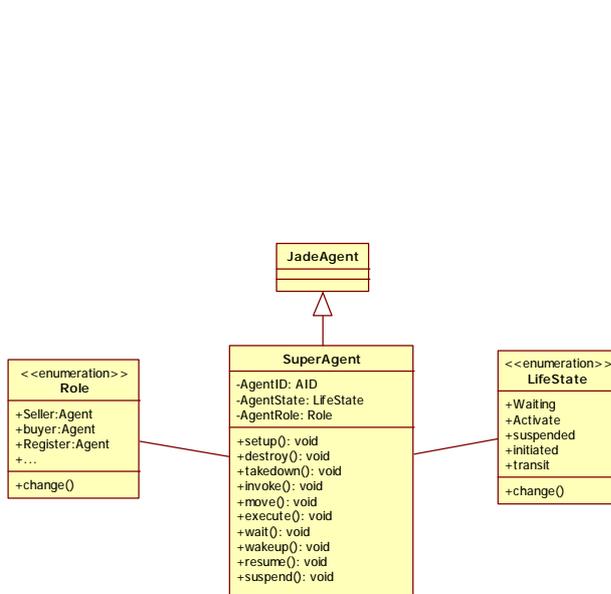


Fig. 2. Design of Agent super class

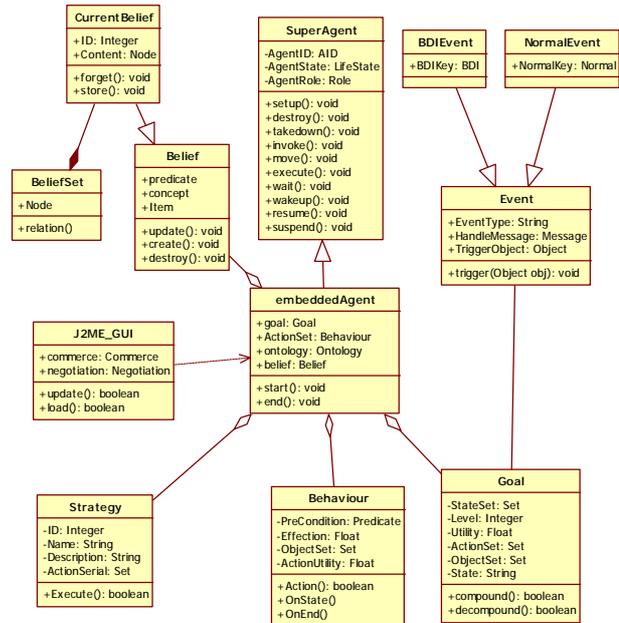


Fig.3. Design class diagram of the system

All Agents class of system inherits from super class. Embedded Agent inherits SuperAgent classes, and includes the properties of goal, ontology, belief, Action set and Action operations, it reflects the Agent specific behavior and state, such as processing message class including sending, receiving, etc., scheduling and executing multiple concurrent tasks of Behavior class. Major Agent categories involved in the design is shown in Fig. 3. BeliefSet class is Agent long-term memory beliefs class, CurrentBelief class is Agent current beliefs class, and they inherits from Belief class. Event class can be derived BDIEvent class and NormalEvent class, is responsible for the management of BDI events and general events. J2ME_GUI class is embedded Agent interface class, is mainly responsible for display Agent user interface. Strategy class is responsible for managing Agent strategy. Goal class is responsible for decomposition and synthesis Agent goals. In order to make Agent have diverse behavior actions, Agent action behavior is further extended, Agent action behavior class designed is shown in Fig. 4.

Whether composite behavior class of CompositeBehaviour or simple behavior class of SimpleBehaviour inherits from the expansion behavior classes of Behaviour. Class of CompositeBehaviour derives series conduct class of SerialBehaviour and parallel behavior class of ParallelBehaviour. Class of SimpleBehaviour derives single simple action class of OneShotBehaviour, cycle behavior class of CyclicBehaviour, behavior classes of TickerBehaviour and WakerBehaviour. Negotiation behavior class of Negotiation and mobile behavior class of MoveBehaviour inherit from series behavior class of SerialBehaviour, and carry on the expansion for negotiation and movement.

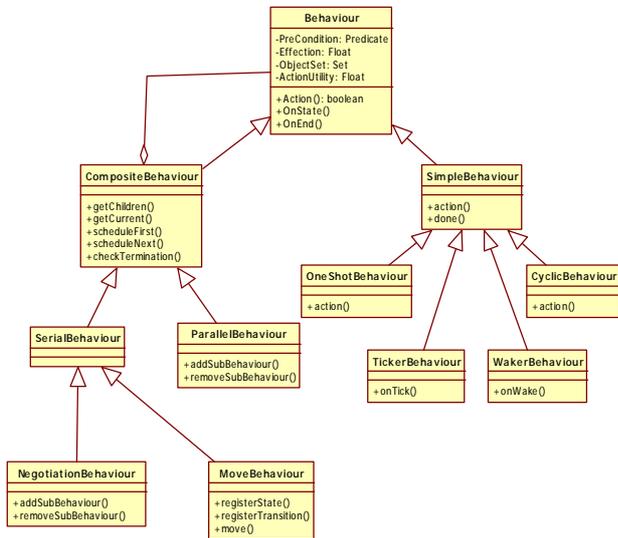


Fig. 4. Agent action behavior class diagram

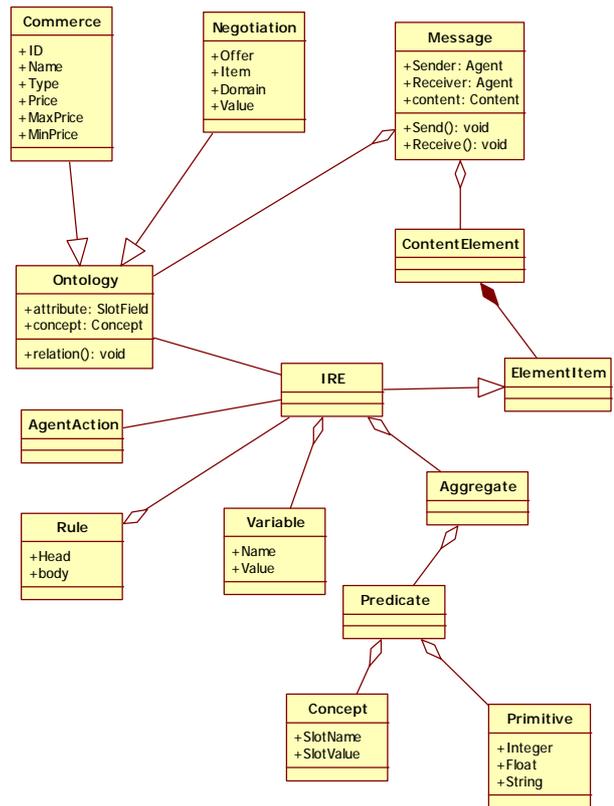


Fig. 5. Message content class diagram

In order to make Agent interactive communication have more flexible semantics, class is designed according to the message involved, as shown in Fig. 5. Assertion class of Predicate is consist of concept class of Concept and primitive type class of Primitive, IRE class is consist of assertion class of Predicate, gathered class of Aggregate, and variable class of Variable. IRE class can be used as rule class of assertion collection organizations and behavior class of AgentAction, also be used as content element class of part of ContentElement, and becomes the part of the message class. Ontology class of Ontology is make up IRE subclass. Commodity class of Commerce and negotiation class of Negotiation of system inherit from ontology class of Ontology. System understands the semantic of the message content by ontology class of Ontology and its subclass. Sometimes the system infers to determine if the behavior preconditions is met to make decisions in accordance with the rule of rule class. The client Agent interfaces are shown in Fig. 6.

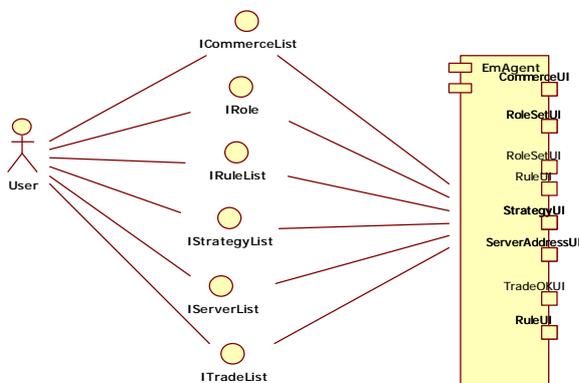


Fig. 6. Client Agent interfaces



Fig. 7. Agent setup interface

As shown in Fig 6. ICommerceList interface provides users to buy goods list function; IRole interface provides Agent roles switching function; IRuleList interface provides edit function for goods filtering rules; IStrategyList interface provides user edit function for bargaining strategy; IServerList

interface provides address list edit function for business server; ITradeList interface provides edit function for transaction confirmation list.

Furthermore , in order to achieve the Agent autonomous reasoning and decision-making function, Agent events, actions, response rules, Agent reasoning in the system, and system overall operation environment are analyzed, and then custom event list, action list, response rules table and response rule base table, facts predicate table, reasoning rules table, reasoning rule base table, description predicate table, action template table, action library table are designed.

Experiment and analysis

PC is regarded as server on which JADE platform is built, and to transplant streamline J2SE system to embedded mobile devices which is regarded as the client program operation platform. To start the JADE main platform on PC, and J2SE system on embedded mobile devices, and on which run the client Agent program, the title of the program is business application assistant, users can open menu, login, see help, browse about information. After the user login system, the embedded client Agent enters operating state, at present the users can use three functions, namely mobile, hang, and setup. If User uses for the first time, it is necessary to carry on basic setup for Agent, setup interface is shown in Fig. 7. In the interface, users can add or edit list of commodity to buy, can also add or edit filtering rules and negotiation strategy, can also use transaction confirmation list to accept or reject the Agent auto-negotiation trading results.

Acknowledgements

This work was supported by the Heilongjiang Nature Science Foundation of China (No. F201212).

Conclusions

The article proposed an Agent model for embedded mobile devices business application, the model is consist of wireless mobile compute environment model, business intelligent Agent general framework, core memory Agent model, analysis the function of each part, system Agents detail design is carried out, and then PC is consider as server on which builds JADE platform, embedded mobile devices is considered as client on which builds streamlined J2SE system, and then the client Agent system program is designed, finally realizes Agent independent mobility and interactive functions between server and client. From the perspective of the operation result of system, the article designed the multi-Agent system realized business applications for embedded mobile devices.

References

- [1] Yong Li, Yiming Chen. Coal Technology. Vol.29 (2011), pp. 183-185. (In Chinese)
- [2] Lei Lei. Research and Design of Embedded mobile Agent Negotiation Platform. Wuhan university of technology, 2011. (In Chinese)
- [3] Jian Zhang, 2012, Study of Agent Role Model and Construction Method of Multi-Agent System, M.S. Thesis, Shandong University. (In Chinese)
- [4] Yingwu Chen, Feng Yao, Jufang Li, Renjie He, Lining Xing. Systems Engineering- Theory & Practice. Vol. 33 (2013), pp. 791-801. (In Chinese)
- [5] Ho N B, Tay J C, Lai E M K. An effective architecture for learning and evolving flexible job-shop schedules. European Journal of Operational Research, vol. 179 (2007), pp. 316-333.
- [6] Michalski R S. Learnable evolution model: Evolution process guided by machine learning. Machine Learning. Vol. 38 (2000), pp.9-40.