Hirashima,T. et al. (Eds.) (2010). Workshop Proceedings of the 18th International Conference on Computers in Education. Putrajaya, Malaysia: Asia-Pacific Society for Computers in Education.

CAMEG – A Multi-Agent Based Context-Aware Mobile Educational Game for On-the-Job Training

Chris LU[a,b], Maiga CHANG[a*], KINSHUK[a], Echo HUANG[b] & Ching-Wen CHEN[b]
[a]*School of Computing and Information Systems, Athabasca University, Canada*
[b]*Dept. of Information Management, Nat'l Kaohsiung First Univ. of Sci. and Tech., Taiwan*
*maigac@athabascau.ca

Abstract: In this paper, we present a multi-agent based context-aware mobile educational game that can generate a series of learning activities for users doing on-the-job training in their working environment. We apply multi-agent architecture (MAA) into the mobile educational game design to achieve the goals of developing a lightweight, flexible, and scalable game on the platform with limited resources such as mobile phones. Multi-agent architecture not only makes different agents have its own tasks, but also provides developers an expandable way to add further functions into the game and to polish agents in order to make improvement on the game. This research focuses on designing the tasks that each agent needs to do and the communications may happen among agents. The benefits of the proposed multi-agent architecture game design makes the game itself easy to maintain and to expand, at meanwhile, reduces computing power consumed by the systems due to not all agents will be needed at same time.

Keywords: Context-aware, knowledge structure; game-based learning; on-the-job training, multi-agent system, situated learning; mobile phone

1. Multi-Agent System and Mobile Educational Game

Mobile phones have limited computing power and resources than desktop and laptop computers, the mobile applications hence are small and simplified (Tan & Kinshuk, 2009). In this research, we propose a mobile educational game under multi-agent architecture to reducing computing power consumed by the mobile learning systems as much as possible.
Intelligent agent is independent computer programs, is capable of acting autonomous and learning continuously to meet its design objectives (Baylor, 1999). Multi-agent system is a system where many agents are living in it. These agents are responsible for their own tasks and collaborate with other agents whose responsibilities belong to the pre- and post-requisite tasks. Researchers have applied multi-agent concept into either learning management system design or mobile educational system design, and report good results in system scalability (Dutchuk, Muhammadi, & Lin, 2009; Zhang & Lin, 2007).

Multi-agent-based system is one of our research's objectives, designing a system with agent-based perspective makes our ubiquitous educational game more flexible and expandable. For instance, the system can find an agent to hold user's playing data temporary if the network is disconnected and the agent asks DB Access agent doing batch update as usual after it detected the network is available again.

The rest of this paper is organized as follows. We first describe the multi-agent based mobile educational game design including system architecture, database design, and the agent collaborations. This paper concludes with the summary and the discussions about possible future directions.

2. Multi-Agent Based Mobile Educational Game

In this research, we use positioning technology and two-dimensional barcode (e.g. QR code) to develop an educational game on mobile phones. To develop a lightweight, flexible, and scalable game, we take multi-agent architecture (MAA) into considerations while designing the educational game. Multi-agent architecture not only makes different agents have different tasks, but also provides us an expandable way to develop further functions, for instances, we can put new agents into the game for special purpose or can replace an old agent with a new and more powerful one. Figure 1 shows our MAA-based system model.

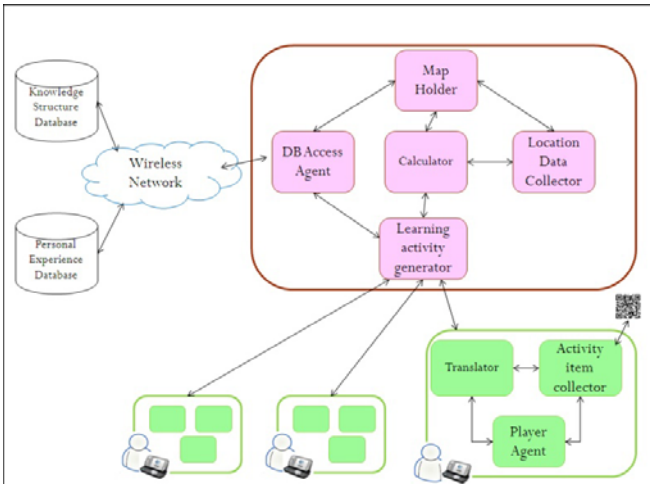

Figure 1. MAA-based system model

The agents in Figure 1 are responsible for specific tasks:

Player - Player agent is a bridge between the user and other agents. It acquires data from other agents such as translator and learning activity generator.

Activity item collector - The main task that activity item collector does is to scan the QR code with the camera and interpret the scanned data.

Translator - Translator can identify different language inputs and store it into database with suitable text encoding method. Translator is very useful in non-English speaking country (e.g. China, India, and Japan) and bilingual environment (e.g. English-French and Dutch-English). Calculator and learning activity generator - The two agents accomplish the tasks for learning activity generation.

Location data collector - Location data collector is responsible for detecting and processing positioning data. If the location data collector is a GPS-based collector, then it gathers the GPS data and interprets the longitudes and latitudes from the data. On the other hand, if the collector is a QR-Code-based collector, it can scan the QR code and interprets the embedded information from the scanned data.

Map holder - Map holder always keeps a copy of the map where the player is playing the game in case any other agents may acquire or the network connection is no longer available. The map here in the game is the context-awareness knowledge structure.

DB access agent - DB access agent makes up the appropriate data manipulation language of SQL commands and receives the results from the database. If the

network connection is not available, the agent will keep the manipulation jobs and do batch update when the network connection is recovered.

To establish such expandable game, we need to design the database with a comprehensive perspective, which means, we need to take knowledge structure, game and role management, authentication, and game-play into consideration. The database design can be explained in four parts, authentication, game, location, and knowledge structure as Figure 2 shows. In order to improve the system flexibility, we use Boyce-Codd normal form (BCNF) database design strategy.

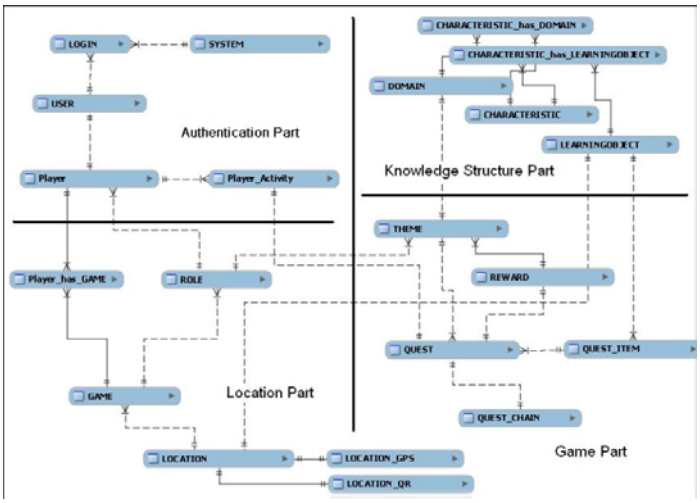

Figure 2. Database view diagram of CAMEG

In authentication part, this game can combine within other learning systems because the authentication relevant tables have user accounts independently from the learning systems and bind the user's ID in the learning system with the user account in the game, for instance, a user can use his/her employee id and password to sign in the game, the game then generates a user account and binds on the employee id when s/he first sign in.

In the game part, we design quest chain, quest, theme, and reward tables. The quest and quest chain tables are built for the learning activity generator in storing the learning activities. We also design rewards for learning activities to motivate users playing the game.

In the location part, we use location table to store learning objects' locations and to provide users right learning activities according to user's location. So the game world can be expanded to cover other areas in the real world easily, such as another building in school campus. In addition, with BCNF, the design of location part allows the game using hybrid positioning technologies freely.

At last, in the knowledge structure part, the main focus is context-awareness. The object table is associated with location, which means every learning object has a location id and can be located in the environment. Moreover, as Figure 2 shows, the knowledge structure has hierarchical characteristics and every learning object has its own characteristics.

Figure 3 illustrates the relations among the agents and database. We have not considered map holder here yet because the game world is only set to the 11th floor of Edmonton Learning Centre, Athabasca University, Canada. Nevertheless, multi-agent-based design lightens the programs, which means we could put a new agent into or replace existing agents anytime later very easily without changing the main program.

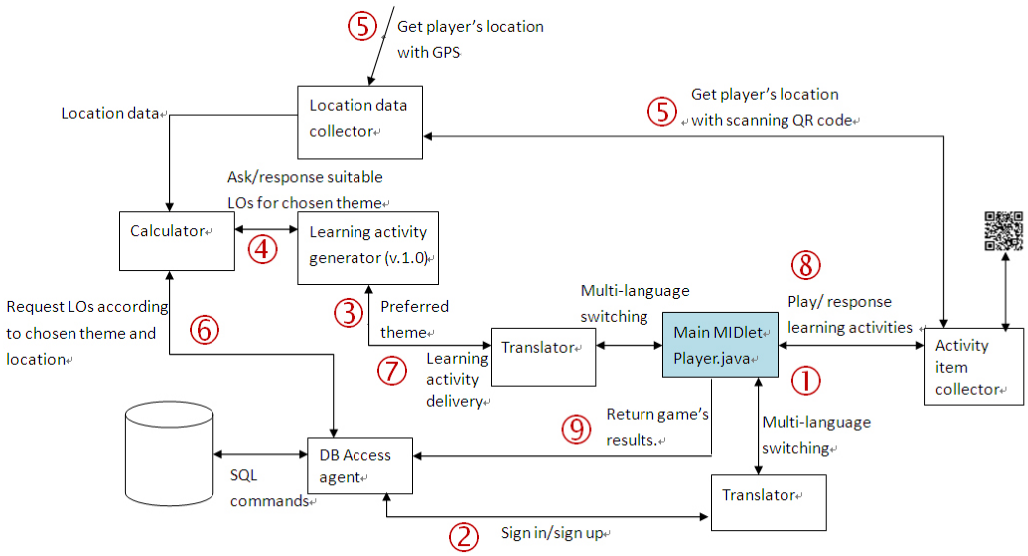

Figure 3. Relations among agents and database

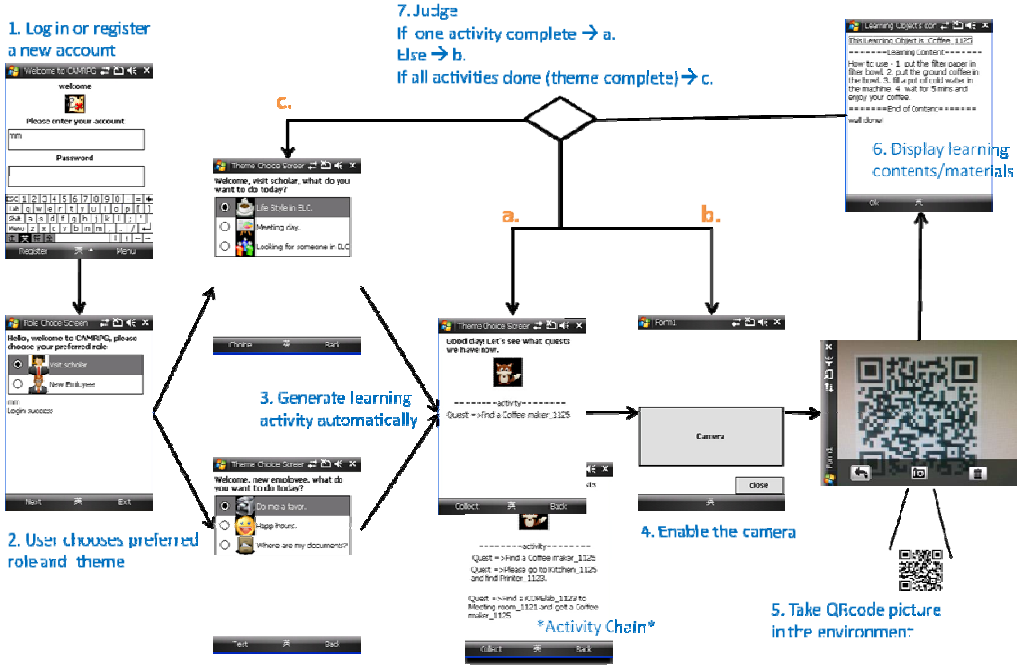

Figure 4. Screen shots of game-playing

We demonstrate the game-play and explain its process from agents' viewpoint. During the game-play, the Player agent will be the only one agent who interacts with the user and helps data exchanges between the user and other agents. At the beginning, the Player agent first gets username and password from the user (as step 1 on Figure 4 shows) and then sends these data to the Translator for checking the language these data uses (as step 1 on Figure 3 shows). The Player agent then sends user's username/password to DB access agent (step 2 on Figure 3). DB access agent judges if the account is existed in either the game database or other system's database (e.g. Moodle database). If the username/password doesn't exist, the Player agent will ask the user to create an account for playing the game.

After the user signed in or registered a new account successfully, the Player agent offers the user two role options, i.e. new employee and visiting scholar, which the user can choose to be (as Step 2 on Figure 4 shows). Each role has some pre-defined themes for user to pick-up. Once the user chose his/her preferred role and theme, the Learning Activity Generator and the Calculator will collaborate with each other to generate corresponding learning activity chain (as Step 4 to 7 on Figure 3 and Step 3 on Figure 4 show).

The user then receives the learning activity one by one offered by the Player agent. These learning activities ask the user to find specific learning objects and collect the learning objects by taking photos on its QR codes. Therefore, the Player agent will help the user to wake the Activity Item Collector up (as Step 8 on Figure 3 shows). The Activity Item Collector enables the built-in camera for the user and decodes the QR code photo that the user took (as Step 4 and 5 on Figure 4 show) Step 6 on Figure 4 shows text-based content as learning material. Beside text-based learning contents, the learning contents can also be designed as HTML-based, binary-based image data, an URL of webpage, media stream and Flash animation to deliver different types of information/knowledge to the user.

At last, the Player agent judges if the collected item is what the learning activity asks for (as Step 7 on Figure 4 shows). In the whole process, these agents are collaborative working to provide the user context-awareness learning activities and mobile game-based learning experiences. The multi-agent based system makes the game easier to design, develop, and alter/replace functions.

3. Summary

In this paper, we present a multi-agent-based mobile educational game. This game can help users doing on-the-job training to get familiar with the new environment; to adapt new working procedures and policies; and, to learn facilities related to their jobs. The proposed game's multi-agent architecture makes itself easy to maintain and to expand, also, reduces computing power consuming due to not all agents will be needed at same time.

The learning activity in this game is boring in some senses, the activity currently only asks the user to find specific learning object. In our next stage, we are going to put narrative elements and storytelling strategies into the learning activity generation process. The story-based learning activity may make users feel they are playing game rather than following orders made by the player agents.

Acknowledgements

The authors wish to acknowledge the support of NSERC, iCORE, Xerox and the research related gift funding provided to the Learning Communities Project by Mr. Allan Markin.

References

[1] Baylor, A. L. (1999). Intelligent agents as cognitive tools for education. *Educational Technology*, *39*(2), 36-40.

[2] Dutchuk, M., Muhammadi, K. A., & Lin, F. (2009). QuizMASter - A multi-agent game-style learning activity. *Proceedings of the 4th International Conference on E-Learning and Games* (Edutainment 2009) (pp. 263-272). Banff, Canada, August 9-11, 2009.

[3] Tan, Q., & Kinshuk. (2009). Client Mobile Software Design Principles for Mobile Learning Systems. *International Journal of Interactive Mobile Technologies*, *3*(1), 32-37.

[4] Zhang, L., & Lin, Q. (2007). MACVE: A mobile agent based framework for large-scale collaborative virtual environments. *Teleoperators & Virtual Environments*, *16*(3), 279-292.