

Application of Distributive Co-operative Design Method in Smart Homes

Ms.V. Kavitha M.E
Senior Lecturer
D.Y.Patil College of Engg., Pune

ABSTRACT

A “smart house” that responds to the dweller’s needs and desires by adjusting lighting, temperature, even ambient music, has reached the millions of homes in the present century. The basic idea of home automation is to employ sensors and control systems to monitor a dwelling, and accordingly adjust the various mechanisms that provide heat, ventilation, lighting, and other services. The work is concerned with the development of a smart home architecture allowing to integrate information from a wide variety of sensors and actuators: information recruited for these elements is processed into microprocessors implementing computational intelligence techniques; cooperative communication between units is implemented through a wireless net into the home; and internet resources allow to link the home with external services. The paper presents an agent-based cooperative design platform which utilizes Web service to realize interoperability in the home appliances for smart homes.

1. Introduction:

With the development of product complexity and precision it becomes more and more difficult for a single person to deal with well. These design and manufacture problems often involve knowledge of multiple aspects and require designers with different specialties to collaborate during the design process. CSCW supported environment can shorten product development period as well as time-to-market. The rapid development of the internet provides a possibility to integrate designers from different place into a cooperative work environment. Embedded system is a kind of application centric computing system designed for some special purpose; this system embeds in other application and performs some special information processing functionalities, and it comprises hardware part and software part. Embedded system has close relation with practical applications. In order to guarantee the design quality and efficiency of the system, it requires not only hardware software co-design, but also cooperative design of experts from different domains.

The paper analyzes the existing Distributed Cooperative Design (DisCoDe) method for embedded system, and have proposed the corresponding design environment for smart homes. The DisCoDe method makes use of mobile agent and Web service technology to realize the distributed cooperation of multi-experts, so experts from different domains are capable of performing embedded system design on different aspect. This is an approach to apply DisCoDe method to the design of smart homes, since in smart homes there is variety of electronic appliances co-ordinated by router or a PC for co-ordination with the outside world.

2. Related Work:

Multi-agent planning and multi-agent coordination have been fully studied in the past years [1] [2]. In [3] a cooperative multi-agent based plant design system whose architecture is a client/server structure is proposed. They present two strategies to improve update speed and solve conflict problem. Researches of interactive mechanism for CSCW at present always tend to utilize multi-agent technology. Liu and his colleagues propose a multi-agent collaborative design system in which human designers and software agents interact with each other [4]. Functional agents act as the roles of designer and manager. In fact, the core of collaboration work is to take full advantage of external resources and services [5]. Cai considers the low efficient of resource re-use for lack of consideration of user’s role, team, design stage, and individuation in design activity [6]. It proposes an agent-based interactive mechanism to build CSCW design activity. The result shows that the system makes design activity more effectively by pushing design resources with a high precision. In [7], three-layer B/S architecture of Browser, Web server and Database is presented. The proposal of the architecture lays a foundation of establishing a flexible and effective cooperative design system under the circumstance of network. Hassan and Elie propose a framework for a multiplatform mobile agent system based on Web services [9]. The framework includes a mobile agent factory, the mobile agents, and the host Web services. It is proposed to design an agent server that is totally based on the ubiquitous technology of Web services to provide a platform-independent framework for realizing the typical capabilities of a mobile agent system. The paper envisages on the intelligence supported by smart homes.[10]

3. Requirements analysis of the cooperative design system

Multi-user cooperative design, distributed design, remote services and cooperative work are very complicated and have put forward new demands for the network application technologies under the circumstances of distribution.

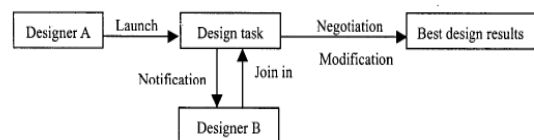


Figure 1.The procedure model of cooperative

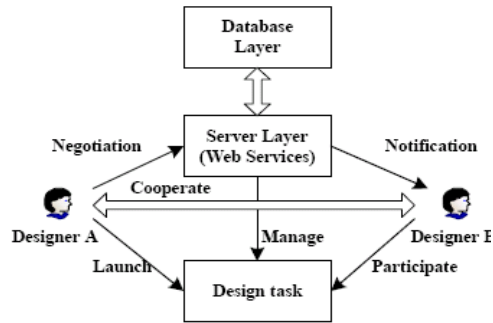


Figure 2 . Another way of representing the procedural model of cooperative

On the foundation of mutual connections and operations, the system should afford support for the client-oriented cooperation, embodies mainly in several aspects as follows:

1. Fine human-computer and human-human interfaces should be ensured. Since the cooperative process is involved with designers of different locations, different sections, and different specialties, the organization structure of cooperative groups is flexible, and their identities of the members of the cooperative groups may span subjects, sections and industries.[12] So in order to further enhance the communication and cooperation between designers, the establishment of a multi-user interface, which is simple and friendly, looks especially important. At the same time, the interfaces provided should be the human-human interfaces in broad sense to guarantee the smoothness of the cooperation. That is to say, firstly multi-display should be supported, and the shared information of cooperation should be displayed vividly in different spots of networks; secondly, the expression of different viewpoints and information should also be supported.
2. Compatibility should be possessed. The capability of transplant and adaptability are of vital importance in the cooperative work under heterogeneous conditions. Clients should cooperatively work together on different hardware platforms, control systems and user interfaces.
3. Strong real-time and practical effect should be ensured. The system should ensure that the cooperative designers share information online and communicate with each other timely and efficiently, and distribute the different network spots, and different clients may use different parts of data during the design process, maintaining the consistency is the crucial point of CSCW. Once a task is accomplished, the cooperative work will be finished with it.
4. The functions of concurrent dealing and control should be provided. As the knowledge domain of each designer is generally independent, the domain knowledge, experience and the capability of solving problems in some degree of each person have the possibility of conflicting with each other, so we need a cooperative framework and conflict resolving mechanism to ensure that all the designers work together to accomplish the common design task.
5. The consistency of product data of the whole system should be assured. Because the product data of design process are distributed in a variety of formats on.

4. The traditional co-operative Design system:

The traditional cooperative design system adopts the ClientServer architecture, which means the system is divided in terms of the tasks based on client and those based on server, making the group cooperative members distributed in time and space. While since the cooperative design is a very complicated process dealing with different information and business, there still remains many unconquerable limitations in this kind of traditional architecture, for example, it lacks support for real-time group interactions, lacks support for the cooperation of multimedia information and lacks transparency towards the control and distributing of system and so on, which results in the low level of expansibility, and also the low capability of opening and being maintained. Based on these deficiencies and combined with the wide applications of network technologies in manufacturing, at present more and more cooperative design systems have began to switch from the traditional C/S architecture to the three-layer B/S architecture. ’

This cooperative services platform based on Browser web Server Database architecture takes commercial browsers as client software and logical server software as Web server and uses commercial data management systems to provide data services. In this architecture the major network management functions and the application software are centralized on the server, thus the distributions on the client end are very simple. Only some browsers and ordinary equipments are required to set up in the reception locations and entrances for each user should be added in the management database.[11]The model simplifies the client software and it only needs to install browsers as the running platform for client applications, while leaving the logical management of the system and applications as well as all the developing, maintenance and upgrading work on the server, which makes the database server offer the storage and maintenance of a great number of structurized data.

The B/S architecture is in favor of allotting computation soundly to different parts of the system, which makes it easy for the integration of system, and once the interface is confirmed, it will not be affected by the changes of each application, and that means all the Web-based application contents is able to use one and the same client software, making all the information from different resources display on the same screen synchronously, which benefits remote design and the cooperation work in manufacturing. When designers send requirements to Web server through browser, the server will deal with the requirements and query database to implement application procedures, and then the server will organize the conclusions as Hyper Text Markup Language pages to be shown on the clients’ browsers and send out them to the designers.

4.1. Client layer

This layer offers application interfaces for designers and it is a graphic client interface. The work of computer is too much heavy because of the dynamic and real-time demands of cooperative design. Since every designer in the cooperative process is updating the design entity, the cooperative design environment should also be updated continuously. In the traditional C/S model system, the client is generally organized by application procedures and corresponding database connectivity procedures, while the server is a certain, kind of database system. With this architecture the client will not only

complete the work of interactions and data display, but also the dealing of operation logic, and as a result the architecture is very huge and complicated and difficult to maintain.[13] So it is a two-layer architecture based on “fat” client. However if the three-layer architecture is adopted, the designers of the client end only need to send the updated information of an entity as messages to the server, which will set up connections with the database system through database server, and accomplish the updating of status via database management technologies, and finally replicate these updated parts to those designers who have the purviews to see the information timely. In this way the work of the client end can largely be simplified and the client can break away from the multifarious labors. The client computer only needs to install a Web browser and some actual design developing software for the designers in this architecture. When designers access the middle-layer design system through browsers, those actual developing software can be integrated with the design tools on the server, with which the functions of the developing software may be expanded. Thus the information communication and sharing between different client ends, the management of design states and also the dispatch and mediation of design process can all be realized.

On the other hand, ActiveX controls can be downloaded from the Web server to be implemented in local places at any moment in line with the requirements. The ActiveX controls are interfaces provided for the methods, attributions and events services demanded by different special business procedures in cooperative design system, which play a magnificent role in softening the load of server when implemented on the client end.

The ActiveX controls are procedure modules with certain functions and are always encapsulated with various powerful functions, such as those complex functions like the query interfaces with Web database, the resource sharing with the server, the interaction operations with local workstation, and also the automatic submission and feedback of data. When the designer’s access pages by means of browsers, the script programs will realize the access and operations of data by transferring the methods of ActiveX controls, and then return the feedback to the server. In addition, the distribution and upgrading of the ActiveX controls are easy to realize. The operations involve only the middle layer distributed on the application server, excluding the client end, making the client really “thin”[8].

4.2. Server layer

The cooperative design between enterprises is a process with which designers communicate with each other and get inspirations from each other, and at the same time it is also a complicated process dealing with a great amount of information and business. The accomplishment of the process demands frequent access and communication between different designers, different systems, and even different modules. Therefore one of the key problems of the cooperative design is to possess a “cooperative environment” which is capable of accomplishing the “common task”, or a certain sharing environment for the cooperative members. This environment mainly takes on network transmitting, notification, information filtration, access control and concurrent control and so on, and it is able to send out various on-the-spot information to all participators, making them know each other’s work condition in time to facilitate the

cooperative work. The realization of the environment relies on setting up a cooperative design server, which is functionally integrated and reliably safe. The server will mostly act as a receiver in the distributed system and is almost like an agent and buffer storage. With the encapsulation of the server, the special functions of the original system can be changed into services that may be transferred by outer clients of remote areas to realize the integration of information, integration of process and integration of function in cooperative design system, and the service access mechanism of distributed environment will then be standardized by a series of recognized criterions. When using the service, the designers merely concern with the results of the services transferred, without considering about the problems of implementation of the services or their material realization.[11] The server layer provides a group of cooperative design toolbox, cooperative management module aggregation, decision-making mechanism and multimedia cooperative application aggregation.

The toolbox includes CAx tools, some ActiveX controls and browser tools. CAx tools are mainly used to integrate with the CAx software, which has already existed on the client end, to extend the functions of the CAx software by making it possess the capability of cooperative design. The browser tools offer the browse and query of design data based on browsers. These diverse tools will facilitate the work of groups of people [9]. The cooperative management module aggregation consists of project management, document management and workflow management. The decision-making mechanism includes knowledge integration, intelligence cooperation and confliction negotiation as well. As for the multimedia cooperative applications, the support of e-mail, videoconferencing, and shared whiteboard are contained. All of the cooperative modules noted above are based on the technologies of STEP, XML, VRML and CSCW. When the distributed cooperative designers organize and launch a cooperative task according to requirements, they can obtain the cooperative tools and function modules from the local or remote servers to communicate with each other. The designers provide needed information for each other while doing their own work.

The client end only needs to do little treatment, storage and sending work for the information to maintain the status consistency of a big virtual environment. Thus the dealing, storage and network requirements of client computers improve with the increase of the entity’s density. The modularization functions allow the clients merely to concern about the results of transfers when using services, without taking into account of either the implementation process of them or the material realization problems, which leads not only to the improvement of the efficiency of information interactions, but also to the assurance of the reliability and integrality of information transfers.

The HTTP transport protocol is adopted as the data exchange protocol. By means of this protocol, when requirements of the client are put forward, the Web server as a middle-layer receives them and submits them to each application server, leaving the groupware to deal with the application requirements of the client layer. By jointly using various kinds information with no gap such as those of the database, hyper text and multimedia, the Web server can offer corresponding services about logic, capability and catalog to accomplish such tasks as the logistic

computation, the query and modification services of data and concurrent management as well, and lastly return the dealing results of to those designers.

4.3. Database layer

This layer provides channels for accessing different data resources. The enterprises participating in the remote cooperative design and manufacturing usually have partial resources to exchange with the outside through network, which requires the server to offer the services of the query and modification of database and concurrent management

and so on. The database layer stores the data in the cooperative design process and accomplishes the business planning, authentication and sustaining storage work together with the Web server layer, so the network-engineering database is the core of the information exchange of cooperative design system. The Web browser connects with the Web server through network protocols, and the server sends out the requirements or data received from the browser to the database, from which it can read relevant text, graphics, illustrations or multimedia information, and then corresponding operations will be implemented to the database on the database layer, and finally the conclusions will be returned to the browser. With the rapid increase of the information amount and the diversification of the data storage, the data structures become more and more complicated, and thus the status of the database layer in cooperative design system becomes more and more important.

Presently there are two structures for the database system in cooperative design system, namely integrated and distributed structure. In the integrated structure there is one network shared database for the storing and obtaining of multi-designer, and the designers can communicate with each other by means of global database and distributed controlling mechanism, while the distributed model ensures the correlative attributes through the effective restrictions of the inner database. [12]

A local restriction manager takes in charge of the requirements related to the local place, and a global one takes in charge of the global restrictions. Of these two models, the integrated structure has more restrictions for every application participating in it while the distributed structure is much more complicated in implementation. To guarantee the independency of each designer and avoid the illegal access or false operations, the system has no necessity and also no capability to open all the data to the external, so this system adopts the database system combining both the integrated structure and distributed one. That is to say every network spot of the cooperative design system has its own local database and is detached with the external network by firewall. A firewall is an agent server, which prevents the computers of the inner network from communicating with those of the outer ones and vice versa. On the other hand a shared database is offered on the server end and the shared data are stored in the shared data center via cooperative design platform, including different kinds of information needed in the cooperative design process, such as the manufacturing resources library, product information library, engineering equipment library, process data library, knowledge library, graph library, model library and so on. The effective unification of the independency and interactions of data are ensured with this kind of multiple database structure.

The three-layer B/S architecture noted above is a kind of application software structure which is capable of spanning

heterogeneous network and computer platforms transparently, and furthermore the little redundancy and multi-user sharing of data are ensured. Since the storage of data is independent of the applications, the security, integrity and concurrent consistency of data are guaranteed too.

5. Conclusion :

According to the actual requirements of the cooperative design system, this paper brings forward a distributed cooperative design architecture based on Web, namely the architecture is composed of the Web browser, Web cooperative sever and shared database, forming a three-layer B/S system for application in smart homes. Combining with the practical characteristics of the cooperative design, the paper analyzes the actual functions of each layer and the relations between them in great detail, pointing out that this three-layer architecture based on groupware insures the usability of system, flexibility of groupware and the capability of maintenance of client end as well.

The cooperative design system needs abundant human-computer interaction manners and real-time cooperative work to accomplish the convenient collection and expression of information, but presently the information interactions and sharing support are far from reaching the ideal cooperative work effect and still need continuous summaries and researches on the cooperative design system in the future so as to build up more models and establish more technologies, and ultimately promote the development of network manufacturing. Thereby making the CSCW system applicable in home automation also. The paper is a proposal for applying in the smart homes and actual implementation is in progress.

6. References

- [1]. Anderson, R.N.: Method for constructing complete annual u.s. life tables. *Vital and Health Stat* 129 (2000) 1–28
- [2]. Erickson, P., Wilson, R., Shannon, I.I.: *Years of Healthy Life*. Number 7 in *Statistical Notes*. US Department of Health and Human Services (1995)
- [3]. Haigh, K.Z., Kiff, L.M., Myers, J., Guralnik, V., Krichbaum, K., Phelps, J., Plocher, T., Toms, D.: *The independent lifestyle assistant (i.l.s.a.): Lessons learned*. Technical Report ACS-P03-023, Honeywell Laboratories, 3660 Technology Drive, Minneapolis, MN 55418 (2003)
- [4]. Warren, S., Craft, R.L., Bosma, J.T.: *Designing smart health care technology into the home of the future*. In: *Workshops on Future Medical Devices: Home Care Technologies for the 21st Century*, Rockville, MD (1999)
- [5]. Krco, S.: *Bluetooth based wireless sensor networks implementation issues and solutions*. In: *Proceedings of Telfor 2002, Belgrade, Serbia and Montenegro* (2002)
- [6]. Haigh, K.Z., Yanco, H.A.: *Automation as caregiver: A survey of issues and Technologies*. In: *AAAI Workshop on 'Automation as caregiver'*, July, American Association for Artificial Intelligence (2002) 39–53
- [7]. Vlassis, N.: *A concise introduction to multiagent systems and distributed AI*, Informatics Institute, University of Amsterdam (2003) <http://www.science.uva.nl/~vlassis/cimasdai>.
- [8]. Gómez, F., Miikkulainen, R.: *Solving non-markovian control tasks with neuroevolution*. In: *Proceedings of the 16th International Joint Conference on Artificial Intelligence*, Denver CO, Morgan Kaufmann (1999) 1356–1361

- [9]. Cecilio Angulo, Ricardo T'ellez, ' Distributed Intelligence for Smart Home Appliances', GREC, Knowledge Engineering Research Group,,Universitat Polit'ecnica de Catalunya, 08800 Vilanova i la Geltr'u, Spain
- [10] W.M. Shen, et al, "An Agent-based Service-oriented Integration Architecture for Collaborative Intelligent Manufacturing", *Robotics and Computer-Integrated anufacturing*, 2007, 23, 315-325.
- [11] Tingting Fu¹, Peng Liu², Yigang Wang¹, and Yehua Du³ , Integrating Agents and Web Services into Cooperative Design Platform of Vehicle Headlights' Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed
- [12] Sikun Li, Zihui Xiong, Tiejun Li' Distributed Cooperative Design Method and Environment for Embedded System' The 9th International Conhence on Computer Supported Cooperative Work *in Design Pfoceediugs*
- [13] JIANG Wei, SHENG Bu-yun, ZHOU Zu-de²' Research on the Architecture of Web-based Distributed Cooperative Design System' The 8th International Conference on Computer Supported Cooperative Work in Design.