# An Adaptable Model for Distributed Collaborative Design

X.L. Chen[1], J.Y.H. Fuh[1], Y.S. Wong[1], Y.Q. Lu[2], W.D. Li[2] and Z.M. Qiu[1]

[1]National University of Singapore, mpefuhyh@nus.edu.sg
[2]Singapore Institute of Manufacturing Technology

**ABSTRACT**

Current engineering design process typically involves more than one designer. In many cases there will be a team of designers participating in the design, along with the corresponding clients. This paper introduces an adaptable collaborative module for Internet-enabled Distributed Computer-Aided Design (DCAD). This module was designed based on a platform-independent model that enables designers to communicate and collaborate each other across different operation systems and applications used in dispersed locations. The adaptable module abstracts basic collaborative functionalities for a group design project and is equipped with web-based project management packages. The main research issues include platform-independency and adaptable structure, distributed design management and coordination, design process representation, collaborative design communication and design visualization. The adaptable model makes the collaborative design plug-in and add-on can be developed flexibly and loaded to different applications running in different operation systems. The proposed adaptable module has been tested and demonstrated with case studies in a distributed collaborative design environment.

**Keywords**: adaptable system framework, distributed design, collaborative design, DCAD.

## 1. INTRODUCTION

When a product is designed through the collective and joint efforts of many designers, the design process can be called as collaborative design. This would include those dispersed functions such as design, manufacturing, assembly, test, quality and purchasing as well as those from suppliers and customers [13]. The main goals of such a collaborative design team might include optimizing the mechanical function of the product, minimizing the production or assembly costs, or ensuring that the product can be easily and economically serviced and maintained [4]. Since a collaborative design team often works in parallels and independently using different engineering tools distributed in separate locations, even across various time zones around the world, the resulting design process may then be called distributed collaborative design. Johansen used time-space 2D matrix to examine cooperative works [5]. The matrix categorizes collaboration into synchronous and asynchronous patterns, shown as Fig. 1. This space-time matrix cannot fully represent the emerging collaboration trends. For example, collaboration may happen among different geographically dispersed companies, or within the same company but between two distributed divisions. Here we extend the matrix to a three-dimensional time-location-group space, defined as O (T, L, G) to describe when, where and who are collaborating.

Compared to the time-space matrix, which is a very useful and concise reference to the particular circumstance, the proposed 3D time-location-group matrix not only looks at whether participants are in the same place, but also whether they are operating at the same time or not. We also consider whether participants are in the same company or group. In the collaboration domain, enterprises are usually only concerned the security of data flow; however, the collaboration systems' architecture and their patterns should also be taken into consideration. Very often, in the domain of one enterprise, the design platforms, such as operation systems, types of network, development tools, database systems, CAD software, etc. are usually heterogeneous, whilst configurations and facilities are often different from company to company. This will result in different architectures and different system functionalities for collaborative design.

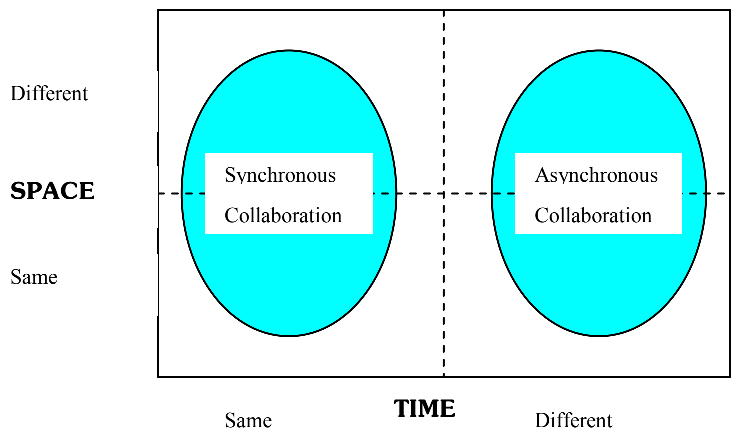


Fig. 1. Johansen Matrix

As shown in Fig. 2, based on the time-coordinate, we define synchronous and asynchronous collaboration. Considering the data location, modelling kernel and functionalities of collaborative design, the tasks can be centralized or distributed. For the participants of collaborative design, it can be inter-enterprise, intra-enterprise or extra-enterprise. Due to the different patterns of collaboration models, it may result in different architectures and solutions for the realization of distributed collaborative design frameworks.

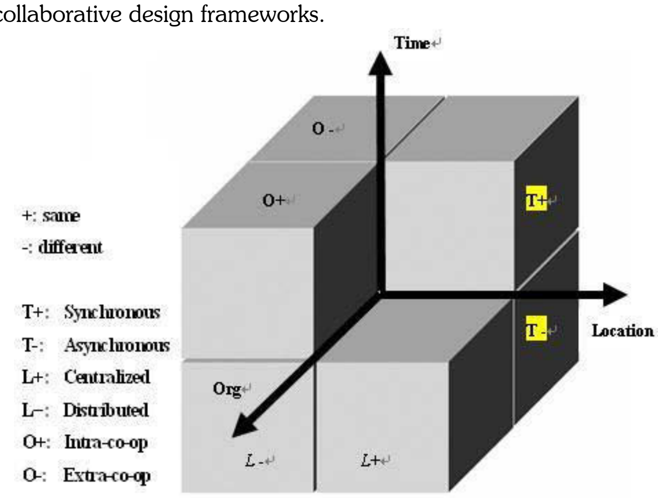


Fig. 2. Co-design Patterns in 3-Dimension

## 2. DCAD FUNDEMENTALS

Collaborative design issues are due to: different groups of people, often of different disciplines, of different enterprises, at different places, having to work together on one product. A single project often comprises of different physical parts and functional modules, i.e. electronic, mechanical, software, hardware and other subsidiary devices. Designing complex products, such as an aircraft or automobile, requires a tremendous collection of expertise, knowledge, technology and tools. Design resources are often distributed. Participants may be in different places as well. Integrated concurrent product development can be realized by leveraging modern information technology to coordinate people, processes, tools and technologies. Smaller engineering companies and medium-sized suppliers are looking for an inexpensive way for geographically dispersed teams to jointly develop products together over the Internet. Traditional CAD/CAE functions nowadays are very often dispersed. Considering the scenario of collaborative design, the key characteristics of collaborative computer aided design compared to the traditional standalone or sequential design process can be summarized as follows: according to the functions and roles of users participating in the design activity, a collaborative CAD can be organized as either a horizontal or a hierarchical manner. The horizontal collaboration emphasizes on allocating a design team from the same discipline to carry out a complex design task simultaneously. The hierarchical collaboration can establish an effective communication channel between upstream design and downstream manufacturing, and it can enrich principles and methodologies of concurrent engineering to link diversified engineering tools dynamically [7].

When those CAD/CAE functions are carried out geographically dispersed for the same project. And those functions may work on different application platforms and operating systems. From the view point of data layer, there are some fundamental problems need to be solved to realize distributed collaborative design, as shown in Fig. 3.

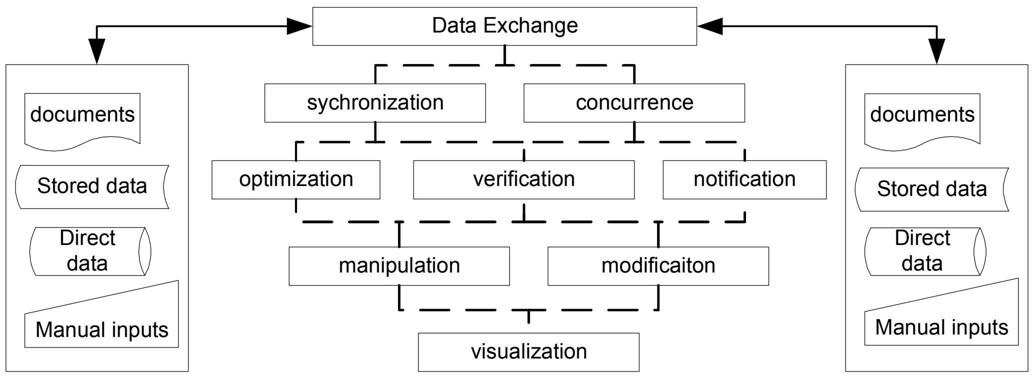


Fig. 3. Data Layer View of Collaborative Functions

Using modern information technologies to realize distributed computer aided collaborative design, the fundamental requirements can be categorized by:
1). how to exchange data of different formats, in different domain definitions, from different locations via different means of communication protocols.
2). how to define and capture behaviours of different people of different disciplines, and organize those user interactions to notify each other. Thus make individual design behaviours in accordance with overall project process.
3). how to develop efficient communication protocols and methods, which makes the content of design data and design behaviour understandable to each other.
4). how to coordinate geographically dispersed groups of involvers, and the decomposed projects and tasks. Thus makes the project process visualized and represented explicitly.
To solve these problems, many researchers as well as commercial software vendors have done much work using different means of methodology, focusing on different issues, and using different tools. In the next section, we will discuss related works carried out regarding these key issues for distributed collaborative design.

## 3. LITERATURE REVIEW

There are three main types of structure in collaborative design systems developed: client-server, agent-based and peer-to-peer. Each has their advantages and disadvantages. And different structures behave differently with different focuses.

Most proposed frameworks use the client-server (C/S) architecture to realize collaborative design systems. Web-based system is a typical implementation of C/S structure, as shown in Fig. 4. The Web is used by the team members as a medium to share data, information and knowledge [16], and in some cases for product data management and project management by integrating the Web with appropriate technologies [10]. In some cases, the Web may only be used to monitor the design process and check the status of the working system [12]. When Java Applets or ActiveX is used for developing the client side user interfaces, clients are getting stronger functions to communicate with the server side as well as other clients. Because when clients query a web page that contains Java Applets or ActiveX object, the web browser will download those necessary components to a local machine first. From this point of view, web is only an access point of the system. WebEx's online meeting system was implemented in such structure. And those cosmos's

VRML viewer is also published as ActiveX component. This makes that there is no much difference between installation-based systems and web-based systems.

Peer-to-peer based architecture allows for a decentralized application design, moving from centralized server models to a distributed model where each peer, independent of software and hardware platforms, can benefit from being connected to millions of other peers. In such architectures, clients and servers have a lateral relationship rather than the traditional vertical relationship, giving the whole peer group tremendous processing power and storage space. The developed architecture, including Begole, et al. [1] and Inventor collaborative tool™, supports the sharing and manipulation of services or modules with other systems. David proposed P2P-based middleware to support content management [3]. Groove Networks is a P2P collaborative software platform that traverses corporate firewalls and allows secure digital collaboration. TOMSCOP is a synchronous P2P collaboration platform over JXTA [14]. And IBM offers grid computing for automotive design. It is worth mentioning that P2P computing is still in an evolving state and much work needs to be done to overcome complex issues such as security, network bandwidth, and architecture designs.

Malone proposes two design principles for agent-based systems, through the experience of developing "intelligent agents" such as Information Lens [9]. There is a growing body of work on agent-based coordination techniques for engineering design. In particular, agent-based collaborative design has been proposed as a promising approach for distributed engineering teams. PACT [2] is an experimental infrastructure for concurrent engineering. The

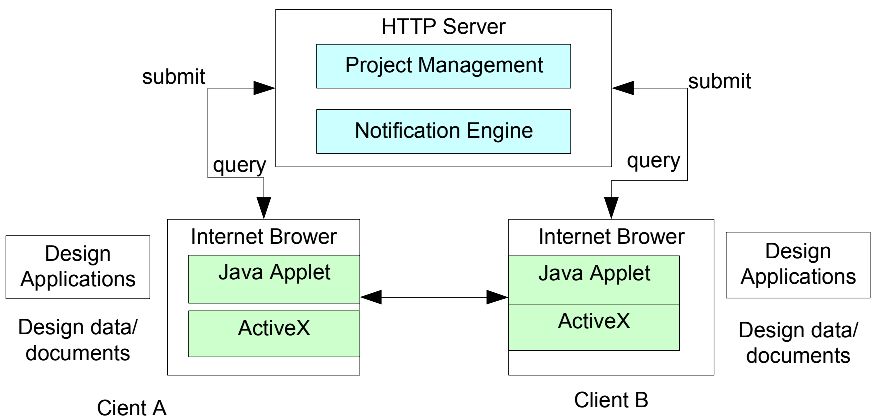


Fig. 4. Typical C/S Structure

architecture is based on interacting agents. A more structured approach to agent coordination is provided by a system called Redux [11].

Different collaborative functions need different communication modes Client-server structure is the most applicable way to implement distributed collaborative design systems. Whereas co-design extended systems, for manufacturing services, CSCW concept level integrations, agent-based structure have more potential values. And peer-to-peer structure is limited in simple functionalities such as application sharing, communication forum, etc.
In the literatures reviewed, there are different frameworks suitable to realize certain collaborative design functionalities. Some researchers proposed new system architectures that include most of the functions in CAD and Co-CAD. In such systems, the collaborative functionalities are often proposed. Unfortunately, those proposed geometric modelling modules are very often difficult to implement, especially when those systems are designed for domain independent collaborative design. It is almost impossible for those end users to implement a solid modelling kernel. To those domain-dependent systems, such as fixture design, bearing design, hydraulic device design, etc., they are easier to be implemented. Even those solutions provided by commercial CAD software vendors, they have their own powerful modelling kernels, such as AutoDesk, EDS, PTC and Co-create, and their collaborative design functions mostly remain on project management and design data exchange level. When they deal with CAD models in other formats, they have to rely on those standard formats, such as STEP, IGES.

A collaborative CAD system cannot be simply set-up through equipping a standalone CAD system with IT and communication facilities. Due to the complexity of collaborative design activities and the specific characteristics/requirements of CAD systems, it needs some innovations or even fundamental changes in many aspects of CAD systems, such as infrastructure design, communication algorithms, geometric computing algorithms, etc. [7]. The followings are conclusions drawn:

Frameworks with an innovative collaborative geometric modelling kernel , such as [6, 7, 8, 15, 8] are difficult to implement. Those with integration functions still remain at distributed projects and design data management level, but they cannot access the design behaviour domain. Solutions provided by current commercial CAD software vendors are strictly restricted to their own software features, and it is difficult for them to provide commercial level interfaces, such as in CollabCAD, OneSpace, AutoCAD, ProE Wildfire.

This research is aimed to solve some of these problems, considering the scenarios of different application platforms in a distributed collaborative design environment and the requirements of design coordination among multiple disciplines. The developed adaptable-distributed-collaborative design module can be customized as plug-ins for possible applications used in the collaborative design environment.

## 4. SYSTEM FRAMEWORK

This paper describes an adaptable architecture for distributed collaborative design and its system implementations, as shown in Fig. 6. DCCAD kernel is abstracted based on the principles of distributed collaborative design, as discussed. The kernel implemented most of the interfaces desired in the realization of collaborative design concept. DCCAD Kernel details will be discussed in the next section. Because most commercial applications provide user development interfaces and customization functions as in the Plug-in or Add-on level, the customized modules are designed in charge of communicating tasks between the application and the DCCAD kernel. Thus the operation behaviours of different applications can be routed through the DCCAD kernel, as shown in Fig. 5.

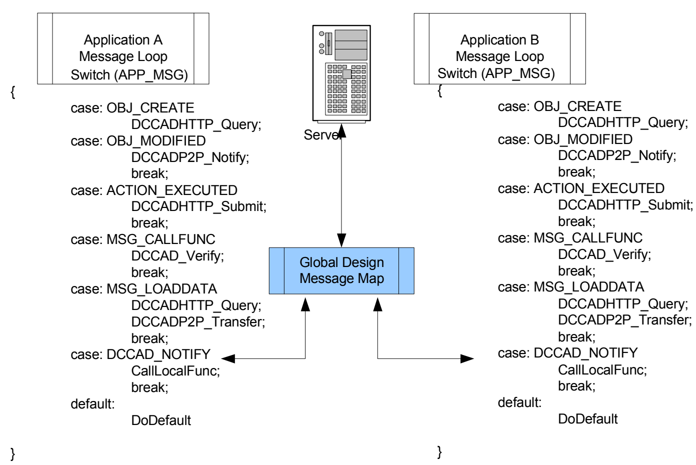


Fig. 5. Design Message Cross Applications

## 5. SYSTEM IMPLEMENTATION

The project management module is designed for distributed project co-ordination and visualization. It uses a task-oriented modeling structure to represent a distributed collaborative design project, with dependency relation tags. The design variable control module is designed to manage global design parameters that have dependency with other tasks. The design management takes responsibilities of general design problems such as design requirement and design process query and updating. The design behaviour control module provides the message-mapping functions. This enables different applications can capture each other's behaviours, such as design is modified, design variable is added, and variable value is set. Communication module provides different communication functions such as HTTP request, socket connection, peer-to-peer connection and message broadcast. Data management module provides interfaces to MYSQL database and XML decoding and encoding functions. These modules function in the name-space of DCCAD kernel and itself has communication protocols and structured data; the implementation structure is shown in Fig. 7.

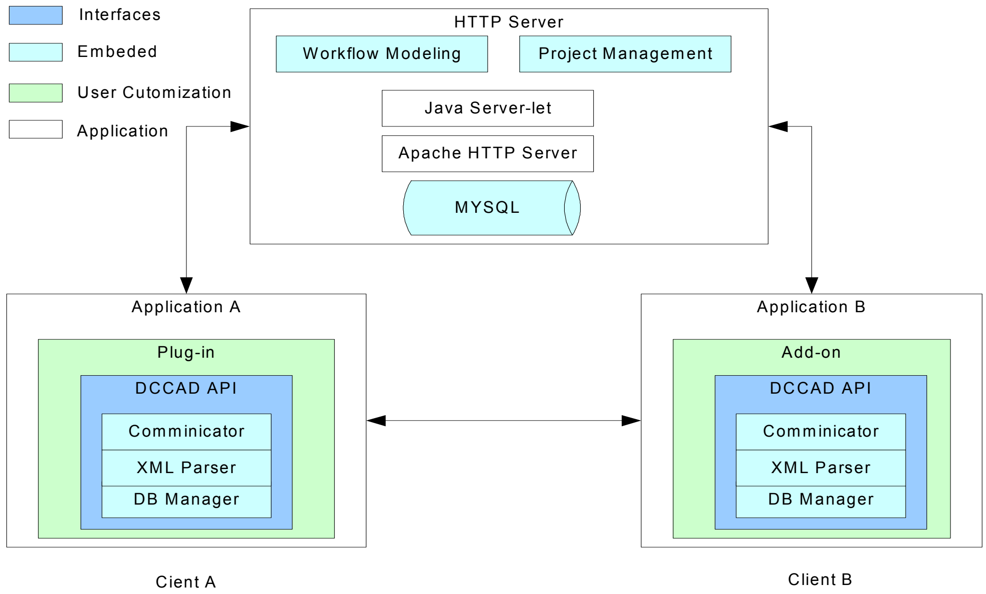


Fig. 6. Adaptable DCCAD Framework

The case assumes that Designer A is designing fixture elements for welding. He uses AutoCAD 2004 as the geometric modelling platform for mechanical structure design. The design specification is composed by Analyst B, who is working on MS Excel 2002 professional, as shown in Fig. 8. When design changes or specification of the design changes, their activities need to be coordinated through a certain mechanism and their design data needs to be expressed and exchanged via certain tools. The following illustrates how the adaptable DCCAD module was constructed and implemented, how the plug-in for AutoCAD and add-on for MS Excel were customized using the DCCAD modules, and how collaborations were fulfilled.

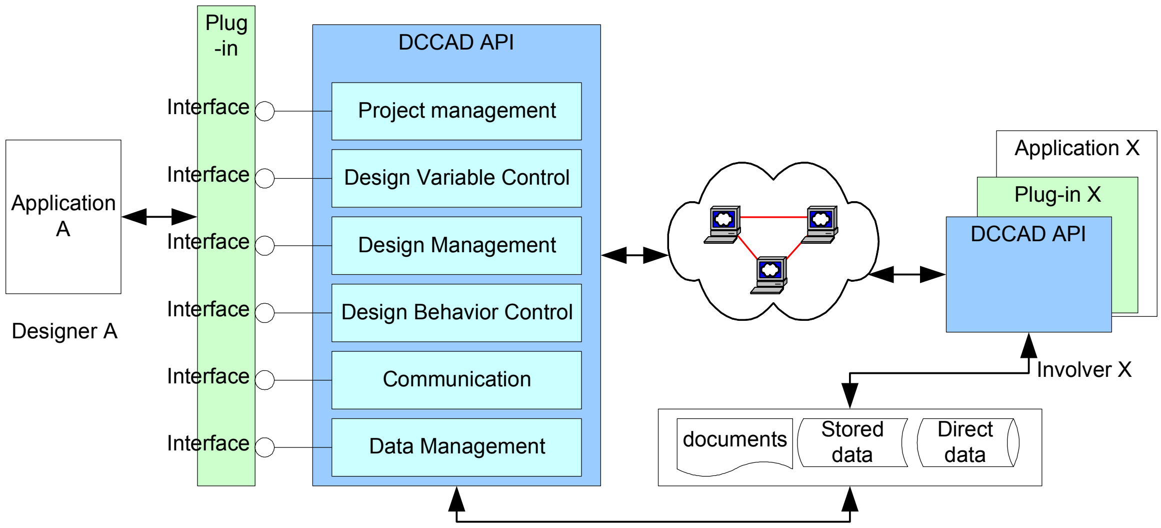


Fig. 7. DCCAD Kernel Functional Structure

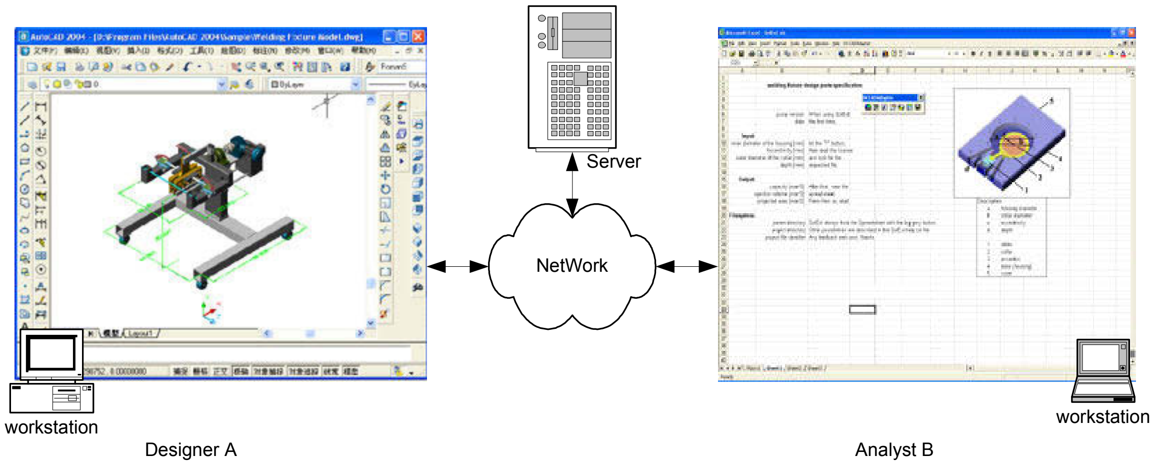


Fig. 8. Co-design Case Scenario

Firstly, plug-ins needs to be implemented for AutoCAD and MS Excel; the implementation structure is shown in Fig. 9. When plug-in is loaded to AutoCAD and MS Excel, the user interfaces will be shown as in Fig. 10. The functionalities depend on the user customization.

Figure 10 shows the analyst working with MS Excel for optimization, based on the design model specifications, design variables and other possible parameters that determine the optimization solution. The analyst binds the design variables to the rule. The design rule will assign the optimized value every time the result comes out. The design engineer can load the optimized or simulated design variables. The next round of design begins until a satisfied result is reached. The designer uses AutoCAD, with the customization based on the adaptable DCCAD; the design variables can be created and bound to the design entities. With the necessary parameters established, optimization and simulation calculations take place using the initial designed values in the system, and the first iteration occurs. In the process of iteration, the analysis system requests to perform the necessary calculations and send the optimized design variables iteratively until the major version stabilized. For a complex product design, design variable quantities might be very huge. The system engineer makes top-level design decisions or modifies the requirements, and other design

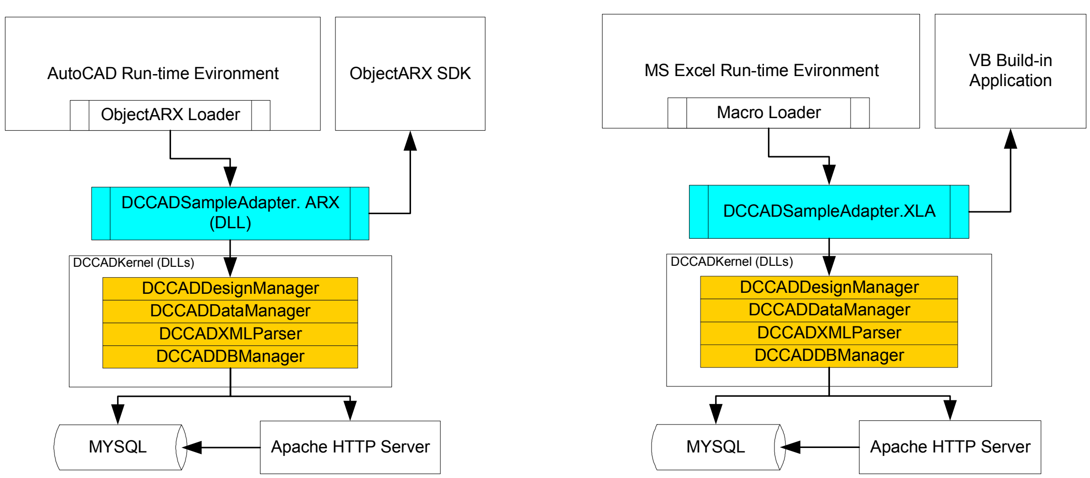


Fig. 9. AutoCAD and MS Excel Plug-in Structure

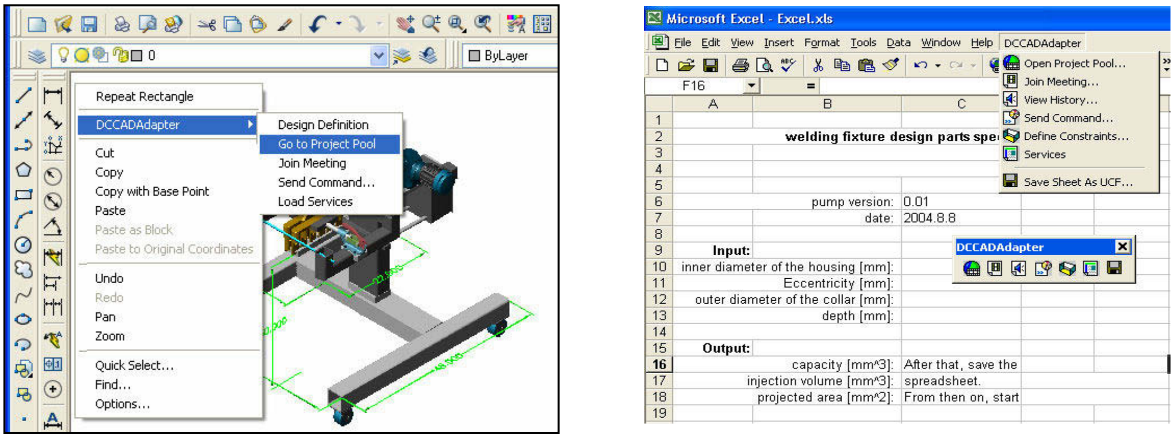


Fig. 10. Customized Plug-in Menus

iteration is performed. The process continues and variants of the design are considered and characterized as determined by the analysis engineer, eventually constituting a thorough examination of the design variations. Figure 11 shows the snapshots of the customized add-in user interfaces.

## 6. CONCLUSION

An adaptable distributed framework, which introduces a new perspective of collaborative design, has been developed to provide a feasible solution. The adaptable module abstracts common functionalities for collaborative design and project management. It would be interesting to carry out more case studies, particularly in different applications involved in the product development process, such as the PLM, PDM system. Further research such as on using agents and grid computing techniques, can be to apply the methodology developed from the studies and to show how the adaptable system would change the distributed collaborative processes. The future work would help support the findings of this research although the development and implementation of these technologies in the actual workplace is still undergoing.

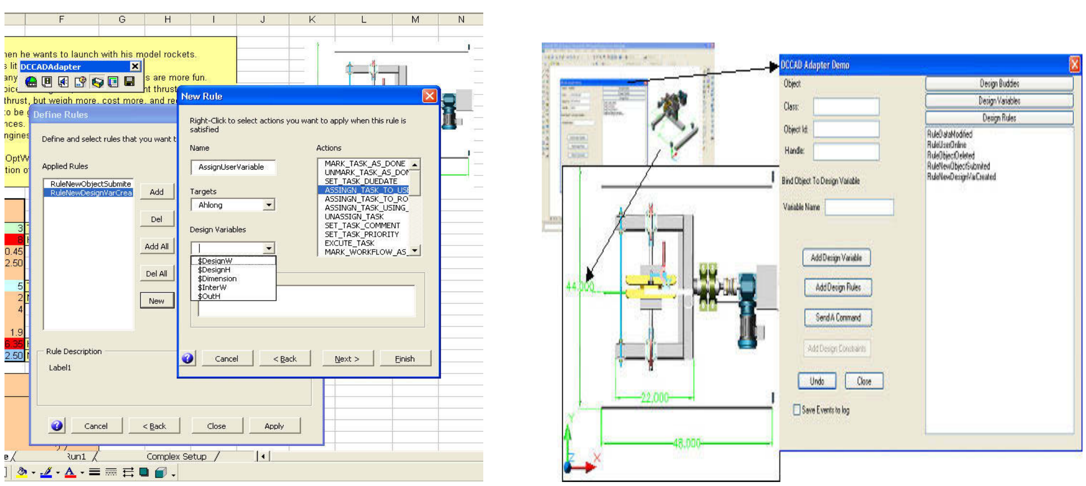


Fig. 11. Snapshots of Customized Plug-in User Interfaces

## 7. REFERENCES

[1] Begole J, Struble CA and Shaffer CA, 1997, Leveraging Java applets: toward collaboration transparency in Java. IEEE Internet Computing, 1-2, pp.57-64.
[2] Cutkosky, M.R., R.S., R.E., Gruber, T.R., Mark, et. al., "PACT: An Experiment in Integrating Concurrent Engineering Systems", Computer, V26, No.1, January 1993, pp28-37.
[3] David Hausheer, Burkhard Stiller: Design of a Distributed P2P-based Content Management Middleware. In Proceedings of 29th Euromicro Conference, IEEE Computer Society Press, Antalya, Turkey, September 1-6, 2003.
[4] Hartly J, Concurrent engineering. Cambridge, MA: Productivity Press, 1992.
[5] Johansen R., "Groupware: Computer Support for Business Teams", The Free Press, New York Johnson, 1998.
[6] Lee J Y, H Kim, S B Han, Web-enabled feature-based modeling in a distributed design environment. Proceeding of DETC99/DFM-8941.
[7] Li W.D, Y.Q. Lu and W.F. Lu, Collaborative Computer-Aided Design – Research and Development Status, Proceeding of CAD2004 at Pattaya, Thailand.
[8] Liu,Q., Tay, F.E.H., Tan, K.C., Kwong, K.K., CyberCAD: Internet Distributed Interactive Collaborative Design, Proceedings of the Internet Workshop 2000, Tsukuba, Japan
[9] Malone, T.W., Lai, K. et al, Agent for information sharing and coordination: A history and some reflections", Software Agents, edited by Bradshaw, J.M, AAAI Press, 109-143, 1997.
[10] Numata, J, Lei, B. et al, Information management for Knowledge Amplification in Virtual Enterprises, IEMC 96.
[11] Petrie, et al, Using Paretooptimality to coordinate distributed agents, AI/EDAM, 9: 269-281, 1995.7
[12] Shen W., J.P. , An experimental environment for exchanging engineering design knowledge by cognitive agents, Proceedings of the IFI, January 1997, Pittsburgh, Pennsylvania, United States.
[13] Sprow E, Chrysler's concurrent engineering challenge, Manufacturing Engineering 1992, 108(4): 35-42.
[14] Tomomi K, J.M., A Synchronous P2P Collaboration Platform over JXTA, 24th International Conference on Distributed Computing Systems Workshops. March 23 - 24, 2004 Hachioji, Tokyo, Japan
[15] Toshiki Mark R. Cutkosky, Design Engineering Technical Conferences September 13-16, 1998, Atlanta, Georgia
[16] Toye G, Cutkosky, M.R., Leifer, LJ et al. SHARE: A methodology and environment for collaborative product development. The Int. J. of lntelligent and Cooperative Inio. 5~yst. 3, 2 (June 1994), 129-53.