Chapter 8
Designing by Services: A New Paradigm for Collaborative Product Development

Hongwei Wang and Heming Zhang

Abstract The design and development of complex products entails the collaborative work of multidisciplinary and geographically distributed teams. The collaborative work largely depends on the effective sharing and integration of information and computing powers in a distributed environment, and thus raises the need of supplying flexible and accessible information for the next generation design systems. Current design systems and tools are mainly focused on specific aspects such as design, analysis, and manufacturing while the need of integrated and collaborative development is not yet addressed. In this research, a paradigm of designing by services is envisaged which is aimed at supporting collaborative product development by integrating information and computing powers provided as services by organizations with different expertise. Such a paradigm requires a flexible architecture and the support of information technologies as it involves a large amount of complex information about products, processes, and people. This book chapter presents a paradigm of designing by services, describes the devising of a service oriented architecture for the design systems for this paradigm, discusses the key enabling technologies involved, and introduces the development of a collaborative simulation using service oriented computing as a case study of software systems implementation.

H. Wang (✉)
School of Engineering, University of Portsmouth, Portsmouth, UK
e-mail: Hongwei.Wang@port.ac.uk

H. Zhang
National CIMS Engineering Research Centre, Tsinghua University, Beijing, China
e-mail: hmz@mail.tsinghua.edu.cn

W. Li and J. Mehnen (eds.), *Cloud Manufacturing*, Springer Series in Advanced Manufacturing, DOI: 10.1007/978-1-4471-4935-4_8,

8.1 Introduction

Engineering design is a complex process in which designers iteratively develop and optimize solutions for technical problems by applying their scientific knowledge. This process is driven by market needs and constrained by available information, human resources, and investment. Wallace identified a number of trends for modern product design and development: increasing globalization; greater competition; shorter life cycles; increasing complexity of processes and products; large distributed design teams; reliance on Information Technology (IT); explosion in available knowledge; and urgent need for sustainable design [1]. As a consequence, the effective management of design processes should be performed to support distributed design teams in developing high-quality, low-cost in terms of both money and time, and sustainable design solutions, and thus raises the need of a significant body of fundamental design research. Some of the popular areas of research include the development of Computer Aided Design (CAD) tools to improve effectiveness and efficiency of producing design solutions, the use of Computer Aided Engineering (CAE) tools and advanced computing to identify design flaws at early stages of the design process to reduce risks, the application of Computer Aided Manufacturing (CAM) tools to speed the transformation of design information into manufacturing information and ease the planning of manufacture, and the management of information/knowledge to improve decision-making by supplying relevant and useful information. These areas are focused on particular aspects of the design work and the resultant tools cannot support large distributed teams. Nevertheless, current research facilitates managing design/manufacturing information electronically and opens up the opportunities for further work supporting distributed design.

Hence, design systems are required for supporting the whole design process rather than only particular aspects such as geometric models and manufacturability analysis. Some studies have been done on studying the next-generation design systems. For instance, Szykman and Sriram indicated that the next-generation product development systems will be distributed and collaborative [2]. A resultant paradigm is called Collaborative Product Development (CPD) which is aimed at developing products by incorporating advantageous resources and making collective decisions amongst distributed teams. A distributed CPD system is particularly helpful for modern product development which is being done more often by geographically and temporally distributed design teams [3]. As a paradigm for supporting the whole design process, CPD involves many issues and requires a large amount of research work. A number of issues have been taken into account in the development of CPD systems, e.g. architecture of information system, team management, communication tools, engineering applications, product geometric representation, integration with CAD/CAE/CAM tools, and knowledge representation [4]. Existent CPD systems mainly focus on supporting activities such as accessing and visualization of design data and design of components and assemblies [4].

The key issues in CPD include system integration and interpretability [3], communication and information representation [4], and project management [4]. In the authors' opinion, there are three challenges for the implementation of CPD. The first one is the sharing of information. Information updated by different teams may be in different formats and have different meanings, which requires the effective integration of information across different infrastructure/systems. In the context of cross organization collaboration, information may need to be kept confidential and only provided as part of a functional service. The second issue is the integration of computing powers as computing is very common in engineering design. This not only requires computing powers to be easily accessible but also involves the coordination of computing processes to perform complex analysis tasks. Thirdly, effective management of the design process is essential as CPD involves various people, processes, and resources. To address these challenges, a flexible and effective architecture is needed and more importantly enormous IT support is required. Hence, provision of data, integration of information, and functional integration of computing resources should all be managed by professional solutions and the technical details involved should be kept away from team members of distributed CPD as much as possible. A new paradigm of Designing By Services (DBS) is proposed to address the challenges, which emphasizes the provision of professional solutions as services and the integration of these services to undertake complex information accessing/integration and the collective use of computing.

The remainder of this chapter is organized as follows. In Sect. 8.2, the literature related to CPD, the next-generation engineering software, and advanced computing technologies is reviewed. In Sect. 8.3, we introduce a new paradigm of DBS for CPD, which is supported by service oriented computing. In Sect. 8.4, key enabling technologies for the proposed paradigm are discussed in detail. In Sect. 8.5, the development of a collaborative simulation system is described as a case study for the implementation of tools and systems for the paradigm. Finally, discussions and conclusions are given in Sect. 8.6.

8.2 Literature Review

Literature reviewed in this research is centered on methods and tools for CPD in a distributed environment. A few keywords can be identified to describe the features of current research on CPD, namely collaboration, integration, information, and computation. Specifically, collaboration is the most typical feature of CPD and involves the methodology of performing development tasks by incorporating the advantageous resources of parties in collaboration. Integration is a key feature of CPD and determines how well the resources can be incorporated in a distributed environment, involving both development methods and computing technologies. Information is dealt with in CPD to better produce, share, use and reuse information and knowledge among parties in collaboration. Lastly, two ways of using computation exist in CPD: the first is the use of computing technologies to support

CPD; the second is to support computation in a distributed environment to perform analysis for CPD. This section is aimed at describing the state-of-the-art CPD research in terms of these four areas.

8.2.1 Distributed Collaborative Product Development

CPD is a methodology of integrated and collaborative design for addressing the requirements raised by the increasingly complex nature of modern product development [5–7]. Hence, system integration and group collaboration can be identified as the key issues in CPD, which have been researched widely [8, 9]. Computer Supported Collaborative Design (CSCD) can be dated back to the 1980s when Computer Supported Cooperative Work (CSCW) became a popular research topic and began to be applied to the engineering domain [10]. In the 1990s, the emergence of Web related technologies greatly promoted CSCD which hence rapidly became a hot topic from the beginning of the 21st century with the support of integration technologies such as CSCW, Web, and agent [10]. Li et al. pointed out that two types of collaboration were mainly found in CPD, namely horizontal collaboration and hierarchical collaboration [8]. The former concerns the cooperation of team members from the same discipline to carry out a complex task in either a synchronous way or an asynchronous way while the latter emphasizes the cooperation between upstream design activities and downstream activities such as manufacturing [8].

Essentially, horizontal collaboration aims to support the collective effort of team members to resolve a joint task. Research on horizontal collaboration can be found in the solving of a number of problems in the design process, e.g. assembly design [11, 12], product modeling [13], and process planning [14]. Major issues of horizontal collaboration include the sharing of information between team members and the licensing of team members to operate on specific pieces of information so as to protect proprietary information and address conflicts. For example, Lee et al. developed a collaborative intelligent CAD framework on the basis of a core design history algorithm that is used to reason about which team member will have the ownership on design models [15]. Hierarchical collaboration, by contrast, is focused on the solving of design problems by team members with different expertise. Research work on hierarchical collaboration can also be found in recent literature. For instance, Park and Seo developed a life-cycle assessment system for evaluating environmental impacts of different design alternatives [16]. Curran et al. proposed an integrated digital design paradigm to facilitate design decision making by taking into consideration manufacturing and cost issues [17]. Apart from the pieces of work mentioned above, other issues of CPD have also attracted the attention of researchers, e.g. interface design and control methodology for CPD tools [18], the use of ontology and semantic technologies for effective and 'meaningful' integration [12, 19], and communication technologies for CPD [20–23].

8.2.2 Knowledge and Information Management

Information representation and sharing is fundamental to CPD as it involves objects and their relationships in a design task. The ultimate goal of representing and sharing information is to make it transferred to, and accessed by authorized users or computer systems, and, clearly, the correctness and effectiveness of CPD is largely determined by it. Li et al. summarized that information that should be shared in CPD involves visualization, products, project management, etc. [8]. Szykman et al. proposed a product model to make design information assessable to users in a Web-based system [24]. The sharing of information involves two levels: the level for system integration and the level for human-understandable information/knowledge. For the latter, a number of integration methods and technologies have been developed, e.g. Web, agent, and CSCW [9]. Shen et al. developed an agent-based integration framework for collaborative intelligent manufacturing [21]. Chu et al. designed and implemented a 3D design environment where information with different levels of details is transferred to different design engineers [25]. Shen and Grafe developed a VR-based virtual prototyping method to support multidisciplinary communication between different systems for performance analysis [26]. Liu et al. proposed a paradigm of composing services to integrate loosely coupled software components which involves detailed design of data and control flows [22].

The sharing of information for information/knowledge management has also been widely studied. For instance, Rodriguez and Al-Ashaab proposed a knowledge driven system architecture to facilitate the provision of knowledge for CPD [4]. Zha and Du developed KSDMME, a Web-based knowledge-intensive distributed module modeling and evaluation framework, to link distributed, heterogeneous knowledge-based design models and tools, and assist designers in evaluation and decision-making [27]. Robin et al. investigated the knowledge exchanged and shared during CPD design processes and identified design context as a key element for supporting design processes and knowledge exchange [28]. It is noteworthy that the two levels of information/knowledge for CPD discussed above do not necessarily mean that current research only focuses on a particular level. In fact most of the work mentioned above was targeted for both of the two levels albeit this was not explicitly stated. Some researchers did make the purpose explicit. For example, Kim et al. developed an ontology-based assembly design method which was aimed at making heterogeneous modeling terms semantically processed by both design collaborators and intelligent systems [12].

8.2.3 Multidisciplinary Simulation and Complex Computation

Simulation technology has been widely applied in industry and plays an increasingly important role in design validation and verification [29, 30], which raises the need to support the simulation and complex computation used in CPD. The development of complex engineering systems, e.g. mechatronics, involves multiple disciplines such as mechanical, electronic, and control, and requires a multidisciplinary approach. The research in multidisciplinary simulation has been proposed by many researchers to go beyond the studies of single-domain simulation [31]. For example, Samin et al. proposed to use Multiphysics modeling and optimization for mechatronic multi-body systems [32]. The use of multiple simulators to perform complex simulation and computation has also been proposed [33], raising the need of studying multidisciplinary collaborative simulation for CPD [6, 29]. In this sense, simulation models and computation resources should be provided as assessable services and integrated at run-time to perform complex simulation tasks. Shen and Grafe developed a VR-based virtual prototyping system for mechatronic system development, which involves the communication between multidisciplinary models [26]. Xiang et al. developed an agent-based composable simulation system for fluid power system [34]. Research has also been done to make computation resources and legacy models/codes assessable as services for effective use and re-use in current/future development tasks. For example, Roselló et al. proposed a component framework to reuse proprietary computer-aided engineering systems/models [35]. Liu et al. proposed a method of composing services to facilitate the integration of loosely coupled software components [22]. Tsai et al. developed a service-oriented modeling and simulation framework for rapid development of distributed applications, which has a focus on the flexible and effective integration of simulation services to build complex applications [36].

8.2.4 Distributed Computing Technologies

The application of distributed computing technologies such as Web, Web Services, Grid, and cloud computing in engineering has been a hot topic since the inception of CPD research due to their powers in distributed communication and system integration [9, 10]. For example, Wang et al. developed a Web/agent-based multidisciplinary design optimization environment [37]; Cheng and Fen developed a Web-based distributed problem-solving environment where computational codes can be accessed and integrated to solve engineering problems [38]; Jiang et al. proposed a Web services and process-view combined approach to the process management in CPD [20]; Fan et al. devised a distributed collaborative design framework and used Peer to Peer (P2P) and Grid technologies for its implementation [39]. Interoperability between CAE software tools is of significant importance to achieve greatly increased benefits in a new generation of product development systems [3]. Lots of work has been done on the system integration and collaboration technologies, e.g. Web, agent, Web Services, and semantic Web, and open standards and commercial tools have already been available [9]. The development of infrastructural technology for CPD systems has also been researched to support distributed integration and collaboration with improved effectiveness and efficiency.

These technologies have also been applied to the development of distributed multidisciplinary simulation systems. Reed et al. developed a Web-based modeling and simulation system which was applied to the aircraft design process and argued that such a system could improve the design process [40]. Wang et al. proposed to develop a multidisciplinary modeling and simulation platform which can support the running and integration of simulation services on the Internet [29]. Byrne et al. reviewed recently research on Web-based Simulation (WBS) and its supporting tools and concluded a number of advantages of WBS, including: easy use; collaboration features; license and deployment models, etc. [41]. One of the important enabling technologies of WBS is middleware which enables different modules in a WBS to interoperate [41]. Some middleware technologies, e.g. CORBA, Web Services and the High Level Architecture (HLA), have been used in developing collaborative simulation systems for engineering design [6, 29]. Among these technologies, Web Services is a very promising technology for system integration on the Web whilst the HLA is an important and heavily researched standard for distributed simulation [41]. Two ways have been identified and researched for the integrated use of Web Services and HLA, namely developing HLA enabling tools using Web Services and making HLA federation to interoperate with other software applications [23, 42, 43].

8.2.5 Discussion

In summary, lots of methods and infrastructural technologies have been developed and applied to support system integration and collaboration for CPD. In particular, four features that have been widely studied can be identified from current CPD research, namely collaboration, integration, information, and computation. However, current research mainly focuses on specific problems in CPD with little work on the development of a widely used tool although it lays a good foundation for further work on CPD by demonstrating that some particular methods are viable and some particular technologies are useful. The potential of CPD is not yet fully explored and new methods/paradigms should be developed and new technologies should be exploited. A number of key issues should be well addressed in future work. Firstly, interoperability plays an important role and shall be well studied to support the integration of complex and heterogeneous systems. Flexibility, effectiveness, and security/privacy are central requirements for system integration. Secondly, a CPD paradigm should allow multiple systems/tools/platforms to work for different purposes and users should be able to select favorite ones [10]. Third, simulation and high-performance computation are essential part of complex product development and should be integrated and supported in CPD. Last but not least, the technical details of collaborative work and system integration should be kept away from users as much as possible. Based on the above discussions, the authors believe that a useful CPD system should be based on the provision of a set of tools/platforms by professional parities and complex development processes should be carried out by selecting part of the professional services. This allows us to envisage a new paradigm of DBS supported by CPD methodology and advanced computing technologies.

8.3 Designing by Services Supported by Service Oriented Computing

8.3.1 Designing by Services

Nowadays, the design and development of products is becoming increasingly complex and requires effective management to ensure the successes of projects. At the corporate level, companies should make strategic plans and provide supportive environment to encourage the application of advanced design methods and the adoption of creative and innovative enterprise cultures. At the project level, necessary resources and relevant personnel should be allocated and systematic design processes should be employed to reduce the number of iterations and improve product quality. Moreover, design tasks are increasingly carried out in a collaborative way due to limited resources and different expertise of individual companies. A scheme of collaborative product development in the Internet distributed environment is shown in Fig. 8.1, where teams with different roles such as designers, managers, and analysts in a project are based in different locations and communicate with each other on the Internet though this is not explicitly shown in the figure. It is noteworthy that each role shown in the figure may also involve several teams rather than just one. For example, designers may come from several teams based in different locations and work collaboratively on the same project.

In the centre of Fig. 8.1 is the iterative design process which is adapted from the classic process proposed by Pahl and Beitz [44]. The input of this process is market needs whilst the output is the definition of detail design solutions. There are several roles identified in CPD, namely marketing people, system analysts, project managers, designers, pattern makers, and others (e.g. finance people and experts on environmental impacts analysis). People with different roles and from different locations work together to make key decisions until optimal solutions have been developed, and in this way, issues about project management, business analysis, manufacturing, environmental impacts, etc. can be addressed in the early stages of design. As mentioned above, different teams may get involved in a single step of

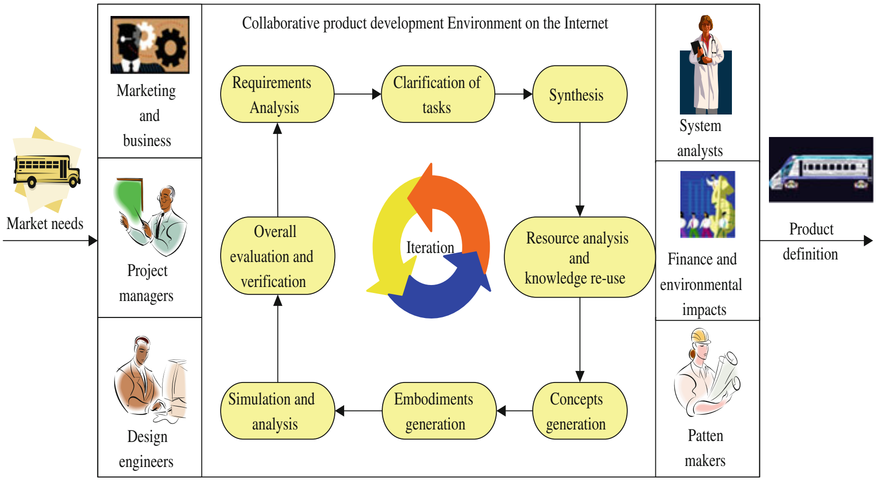

Fig. 8.1 A scheme of collaborative product development

the process. For example, concepts/embodiments generation can be done by teams with expertise on different disciplines such as mechanical, control, and hydraulics. Apart from the core design steps such as clarification of tasks and concepts/embodiments generation, other key tasks in complex product design are also included, e.g. requirement analysis, simulation, and knowledge re-use. The typical features of CPD discussed in Sect. 8.2 are reflected in the process. Firstly, collaboration is required for nearly all the tasks. Second, integration is clearly desirable as information sharing and exchange is essential in the scheme. Third, information is important for supporting analysis, synthesis, and project management. Lastly, computation can also be used in other tasks such as computational synthesis and simulation analysis.

A number of CPD characteristics can be identified from the scheme. Firstly, it involves a large amount of information which needs to be shared and exchanged among team members. Second, it depends on the use of various tools such as CAD packages and information management tools, and requires great IT support for team collaboration and system integration. Third, it concerns lots of professional knowledge, e.g. simulation and environmental impacts analysis. Fourth, the scheme involves an iterative process and requires seamless integration between different stages, raising the need for process management. Moreover, privacy and security is a key issue in this scheme as the sharing of information may be cautious. Therefore, lots of tasks need to be carried out by professional teams to undertake collaboration and accomplish design objectives. These tasks should be made clear at the beginning of a project and the design system should be flexible, effective, and secure. It is thus useful for different parties in collaboration to specify what kind of services they can offer and more importantly a professional team should work out the integration of these services through integration services. As such a paradigm can be envisaged to undertake complex design and development tasks by finding and using professional services which enable effective integration and allow team members only focus their portion of work. Examples of these services include services for producing design, services for supplying requirement information, services for computation, and services for complex integration.

8.3.2 A Service-Oriented System Architecture

As discussed in the last section, the DBS paradigm depends on various services provided by parties in collaboration. These services are developed and maintained by parties with relevant professional knowledge, operate in different locations, and will ultimately be integrated in a design system which offers graphical interfaces for users to carry out design tasks. Such a paradigm actually entails lots of research work and two issues need to be addressed in the first instance for its successful implementation. The first issue is the identification of various services to make clear division of tasks and achieve cost-effective system integration. This issue thus requires an analysis of the roles and tasks in the scheme shown in Fig. 8.1. The second issue is the computing technologies to be employed to underpin the paradigm, which greatly influences the performance of design systems. This section is aimed at understanding the services needed for CPD and their implementation technologies.

A service-oriented architecture is shown in Fig. 8.2, which offers a reference framework for the software systems for DBS. Broadly, any task performing provision of information, computation, and integration of any two processes can be viewed as a service in CPD. In the proposed DBS paradigm, there are eight kinds of services as shown in Fig. 8.2. Information services involve the processing and provision of information such as market needs, design specification, and manufacturing resources, which are aimed at facilitating decision-making in design. This research distinguishes knowledge services and information services in that the former are mainly used to capture, store, and supply pieces of design knowledge. In this sense, knowledge is viewed as a kind of intellectual property rather than plain information, e.g. how to process materials and why a component was designed in the way it is. Process services are used to manage the configuration and execution of complex CPD processes. Integration services are mainly used to facilitate information integration between two systems and manage possible changes in the systems involved. Security services, as the name suggests, mainly deal with security issues. Security issues are not only about the protection of information but also concern the cautious sharing of information between different parties in collaboration. Design services are developed to facilitate the execution of design tasks such as concepts generation and evaluation, drawing production, and synonymous operation on CAD models. Simulation services specifically refer to the components in multidisciplinary simulation for complex problems, which are integrated at run-time to evaluate the performance of a

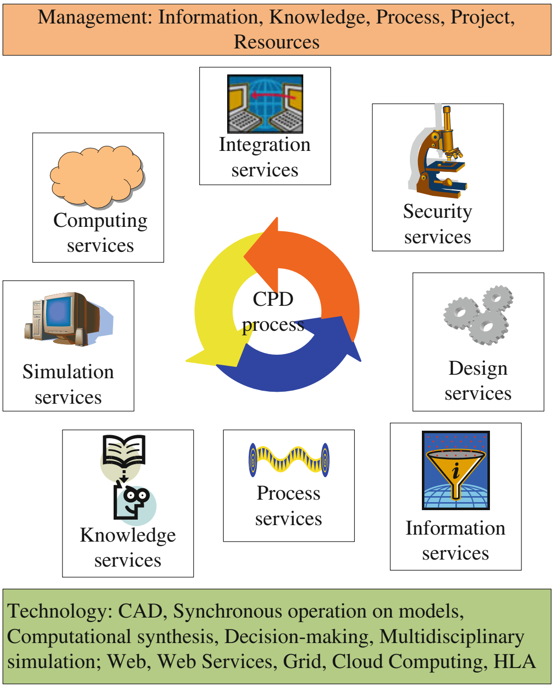

Fig. 8.2 A service-oriented architecture for designing by services

proposed solution at the system level. Some computations used in the design process are also provided as simulation services, e.g. computational synthesis and reasoning on design data.

8.3.3 A Scenario of Designing by Services

To explain how the service-oriented architecture supports DBS, a scenario is illustrated as follows on the basis of the key stages of CPD. Tasks in the design process and relevant services involved in each individual task are shown in Table 8.1 where words in brackets are used to describe the purposes and functions of the services.

It is noteworthy that process services are actually needed in nearly all the tasks and they are only shown in the first task for the sake of brevity. At the beginning of the design process, team members interact with a design system which then starts the process through process services and grants members different accessing rights. The initialization of computing services and integration services is also done at this stage. At the stages of requirements analysis and tasks clarification, information services are mainly used to facilitate decision-making and store important information such as design specification and project management for

Table 8.1 CPD process stages supported by services

Design tasks	Services involved in the tasks
Starting the design process and joining virtual working environment	Process services (monitoring and controlling the whole process), security services (licensing of access and granting of ownership), computing services (establishment of distributed working), and integration services (sending relevant information to other systems)
Requirements analysis	Information services (provision of requirement information for decision making)
Clarification of tasks	Information services (recording design specification and informing relevant members)
Synthesis	Simulation services (computation), Knowledge services (supply of knowledge for understanding problems and making decisions), and Integration services (sending results of synthesis over to design services)
Resource analysis and knowledge re-use	Information services (information about resources) and knowledge services (knowledge about previous projects)
Concepts and embodiments generation	Design services (supporting the creation of solutions and production of drawings), information services (provision of relevant in information response to queries), knowledge services (supply of knowledge about specific topics and previous projects in response to queries), integration services (supporting collaborative work and sending design information for simulation)
Simulation analysis	Simulation services (creation and encapsulation of simulation models, running of multidisciplinary simulation), integration services (integration with design services and sending results for overall evaluation)
Overall evaluation and verification	Information services (provision of relevant information), simulation services (obtaining and analysing simulation results), and integration services (sending evaluation results to relevant systems)

later usage. For synthesis, computation may be used to explore the design space and thus simulation services are involved. In addition, synthesis also requires relevant knowledge for designers to understand design problems and generate ideas whilst the resultant synthesis information needs to be kept for use at later stages. Therefore, knowledge and integration services are also needed. As the name suggests, resource analysis and knowledge re-use requires information/ knowledge services. Concepts/embodiments generation is the core of the design process and involves design services and other supporting services for information, knowledge, and system integration. Clearly, simulation services are needed for simulation analysis for which integration services are also essential as simulation results should be sent to other systems for further analysis and comparison. The last task of overall evaluation depends on simulation results and sufficient information about manufacturing, environmental impact, etc., and thus should be supported by simulation services, information services, and integration services.

8.4 Key Enabling Technologies for Designing by Services

A paradigm of DBS is described in the last section, with eight kinds of services identified and a service-oriented architecture developed. A scenario is also introduced to illustrate how the services are used in the design process. The eight services have different focuses and clearly require the support of many technologies which not only concern product design and development but can also include distributed computing technologies. In this section, a few enabling technologies will be discussed to underpin the implementation of software systems for DBS, as well as to identify promising directions for further research. These technologies are discussed in detail in the remainder of this section.

8.4.1 Distributed Computing Technologies

Apart from the eight kinds of services, management and technology are two important issues in the service-oriented architecture. On the one hand, the development of this architecture is aimed at supporting effective management of CPD and thus the management of information, knowledge, resources, processes, and projects should be taken into account in the design of services. On the other hand, technologies are also important for CPD in a distributed environment. Some specified design technologies such as CAD, decision-making, computational synthesis, and multidisciplinary simulation should be well considered. Moreover, distributed computing technologies should also be utilized to support the service-oriented architecture. In this research, Web, Web Services, Grid, Cloud computing, and the HLA are identified as promising technologies. Specifically, the Web can be used to implement Web-based system which offers a virtual working environment for CPD. Web Services can be employed in a number of ways, e.g. development of information and knowledge services, implementation of process and integration services, and encapsulation of simulation and design services. The Grid can be used for sharing computation resources (i.e. simulation services) and supporting provision of design services. Cloud computing can be applied for massive storage of information/knowledge, implementation of security services, and the shared use of tools/systems/platforms. These technologies have different features and powers, and in practice they should be chosen based on system requirements and available budgets. Detailed discussions on these technologies can be found in [9, 41].

8.4.2 Design Data Exchange and Synchronous Operation of Designers

CAD tools are widely used in engineering design, which are increasingly powerful and can improve the effectiveness and efficiency of model creation and drawing production. Modern CAD packages may also have advanced functionalities such as kinematic analysis, knowledge capture and re-use, and integration with CAM tools. In CPD, design solutions are created collaboratively by distributed team members who may use different CAD packages. Moreover, synchronous operation on CAD models is required when several teams work on the same component/assembly. Therefore, design services need to address two issues. The first is the use and integration of multiple CAD tools and the exchange and management of design data. CAD packages provided by different vendors may rely on different methods for geometric information representation, raising the need of studying the exchange of CAD models among different teams without loss of key design information. This means that a function of design services is to obtain design information from one site and accurately transform it into a data format which can be accepted by another site. Some research has been done in this area but further work, nevertheless, is still necessary to deal with complex information.

The second issue is the synchronous operation on design models. This is critical for online collaboration in which several teams work on the same model. The key to this issue is the granting of ownership to a team, i.e. only one team can operate on a model at a short period of time. If all the teams are waiting for each other, the system will go into dead lock. Moreover, if some teams have to wait for a very long period, the performance of collaborative work is also degraded. Design services thus need to implement the management of design models to support synchronous operations. This piece of work may also be needed by whichever of the other services (e.g. collaborative simulation) that involves synchronous operations on system models and information objects.

8.4.3 Information and Knowledge Management

As discussion in Sect. 8.3.3, information services are very important for CPD and appear in many stages of the design process whilst knowledge services are also necessary for some key stages. In complex design projects, designers need a large amount of information to generate ideas, evaluate solutions, and make decisions. This is more prominent in the context of CPD in which these tasks are often carried out by several teams. Engineering designers actually have various knowledge needs and the supply of relevant knowledge is very helpful [45]. Information and knowledge management is thus of significant importance for DBS. To make informed decisions, designers need to obtain information about market needs, project management, manufacturing resources, and environmental impacts. In the service-oriented architecture, information services need to support the capture of both structured and unstructured information and supply the captured information to team members based on their context of working. Moreover, the sharing of information should be limited to the extent that confidential information of any individual team is kept safe. The issues discussed above can all determine the implementation of information services.

Similarly, knowledge should be shared 'cautiously' as well because it is perceived as a kind of intellectual property and largely determines the competitive strength of an organization. Knowledge services need to deal with the licensing of knowledge access as well as to support the capture and store of knowledge as design projects proceed. It is a key issue that the capture of knowledge should not obstruct the work of designers and project managers and as such, it is also necessary to develop effective knowledge capture methods. In CPD, design tasks are carried by several distributed teams that have different methods for knowledge management, which makes the capture of knowledge even more difficult. In addition to the capture and storage of knowledge, its subsequent retrieval and re-use is also a critical issue which has not been well addressed [46]. In the context of distributed CPD, the retrieval of knowledge becomes more complicated as knowledge records may be stored in different places and the knowledge needs of different team members can be diverse. These issues need to be well addressed by the knowledge services.

8.4.4 Information Retrieval and Semantic Technology

As discussed in the last section, the subsequent retrieval and re-use of design knowledge is a critical issue. Actually, various retrieval methods need to be developed for the proposed DBS paradigm which concerns information in various forms. For example, design knowledge is generally captured in a structured way and as such its retrieval methods can exploit the inherent structure to improve retrieval performance [46]. Simulation technology is widely used in complex product development and the number of simulation models is large in CPD. Retrieval methods are required for these models to support engineers to find models as per specific needs such as function, discipline, and performance. Information retrieval is a popular area of research and has developed rapidly in recent years due to the emergence of commercial searching engines. Many methods have been developed, e.g. keyword-based search, language models, and machine learning techniques [47]. Apart from the technologies and methods mentioned above, the information seeking behaviours of designers should also be taken into account in the development of effective retrieval methods [48].

The development of retrieval methods depends on the specific structure, format and content of information as well as the information needs of engineers. These issues all need to be considered in practice. There are two promising areas of research for the retrieval of design information. The first is raised by the fact that design engineers often cannot express their needs explicitly and re-phrasing is common in face-to-face knowledge acquisition. Therefore, it is useful to develop a system which can 'chat' to designers like an experienced colleague to clarify information needs, and as such supply more relevant information. The second area is Semantic technology which is not only promising in information retrieval but can also help improve the effectiveness of system integration. In this section, only its potential use in design information retrieval is discussed. With the support of Semantic technology, retrieval methods can better understand design engineers' queries whilst being 'aware' of the contents of information. Thus improved retrieval performance can be achieved and intelligent assistance can be offered to designers. A lot of work can be done in this area as information records accumulate.

8.4.5 Multidisciplinary Collaborative Simulation

Multidisciplinary Collaborative Simulation (MCS) is an important technology for both CPD and Virtual Prototyping (VP), aiming at evaluating the performance of design solutions at the system level and using a digital scheme. MCS achieves rapid development in recent years ascribed to the wide application of CAE tools, with an emphasis on the synergic collaboration of multidisciplinary computational models at simulation run-time. There are mainly two methods for MCS: a centralized method and a distributed method. The former is generally implemented based on the programming interfaces between simulation tools and selects one of the models as the central model that communicates with others at run-time using the interfaces. The distributed method emphasizes a more general framework in which an external programme is used to communicate with all the models. As well as being flexible and scalable, the distributed method also has advantages of supporting model integration in a distributed environment (i.e. good accessibility) and enabling cautious sharing of information. A manifest drawback of the centralized method is that it is only applicable to the cases where the simulations tools involved have interfaces between each other. The challenges for the distributed method include the modeling of complex systems, the run-time assembly between computational models with different integration methods, the 'cautious' sharing of simulation data, and the effectiveness and efficiency of data transfer in a distributed environment. Simulation services are an important part of the service-oriented architecture and therefore should be included by any design system for DBS. The use of virtual simulation can greatly improve effectiveness and efficiency of system evaluation and reduce evaluation costs in terms of both money and time. The challenges are also prominent, e.g. how to improve simulation performance in terms of accuracy, speed of running, and reliability. So far, successful applications of MCS to design problems are not many, and thus the development of effective MCS systems and the improvement of performance are promising areas for further work.

Apart from the enabling technologies discussed above, there are also other interesting and important topics, e.g. the development of straightforward interfaces for design engineers and the accessing of design system from mobile devices. Moreover, seamless integration between design and other life-cycle issues, such as service, disposal, and inventory analysis, also deserves further work. These issues can all be taken into account in the implementation of a DBS system on the basis of its specific requirements.

8.5 Service Oriented Collaborative Simulation

A DBS paradigm and its enabling technologies have been discussed in the last two sections. Such a paradigm can support the complex CPD process by developing a modular and flexible architecture. Tasks requiring great expertise, such as model creation, system integration, information/knowledge management, multidisciplinary simulation, and system evaluation, are all carried out by assessing relevant services supplied by specialized teams. In this way, designers and project managers, as users of a design system, can work in a virtual environment to complete design tasks by assessing and integrating services without the need of fully understanding the technical details. The integrated use of services is flexible and facilitates the re-use of both infrastructure and expertise. The implementation of this paradigm is actually complicated and depends upon lots of technologies, raising the need for understanding the requirements and methodologies for developing design systems. In this section, the design and implementation of a service-oriented collaborative simulation system is described as a case study for developing design systems for DBS.

8.5.1 System Analysis and Design

Multidisciplinary collaborative simulation is a central part of DBS as it is aimed at supporting the collaborative development of simulation models and the running of simulation for evaluating design solutions at the system level. In collaborative simulation, several teams work at different locations to create models, establish communications, run simulations, and analyze results obtained from simulations. It is a typical case of CPD albeit it does not cover all components of the architecture discussed in Sect. 8.3.2. Firstly, it involves all the four issues of CPD, namely, information, collaboration, integration, and computation. Secondly, it also requires a flexible system framework and involves various functions to be provided as services. Thirdly, it needs to be supported by distributed computing technologies. Therefore, the issues considered in its design are similar to those involved in DBS.

Computation is the core of multidisciplinary collaborative simulation as simulation models run in parallel and exchange data to complete complex simulation tasks. Hence, simulation models need to be provided as services which can effectively handle data exchange during run-time and hide technical details by providing only functional interfaces. Information services are also important as they can supply useful information about both the simulation problem and the simulation process. Moreover, engineers should be able to access simulation data to identify problems and develop improved solutions. The use of simulation is aimed at evaluating performance of proposed solutions and therefore the accuracy of simulation is not only determined by numerical algorithms but also largely depends on the accurate transfer of design data, making the function of integration services prominent. Collaboration is an important feature of collaborative simulation as it is needed for many different stages such as problem formulation, system analysis, model creation, simulation running, and decision-making. Simulation models are often created based on CAD models and as such the discussion on data exchange in Sect. 8.4.2 is still applicable and the requirement for synchronous operation on system models is still demanded. Collaborative simulation also requires distributed computing technology whilst having a focus on the accuracy of data transfer in terms of both time order and variable value. In summary, a flexible service-oriented solution is required for collaborative simulation.

8.5.2 A System Framework for Multidisciplinary Collaborative Simulation

A model-centric framework for multidisciplinary collaborative simulation system is proposed by the authors, as shown in Fig. 8.3. This simulation-based design approach is specifically applied for the transition between conceptual design and embodiment generation. At this stage, many decisions need to be made and simulation can be used to provide important information about the performance of proposed solutions. The framework consists of three main tiers, namely application tier, platform tier, and infrastructure tier. Application tier is on the top and consists of the functions that a software system should provide for product development. There are three core components for the system, namely multidisciplinary simulation, multidisciplinary optimization and Product Data Management (PDM). Simulation is carried out in an integrated modeling and simulation environment and simulation results are transferred to an optimization engine in real-time. Optimized variables are then sent back to the simulation environment to execute the next step of simulation. The models and data after simulation and optimization are managed in the PDM system for further usage.

The platform tier comprises the enabling tools for system modeling, simulation, analysis and optimization. Domain models in product design are generally

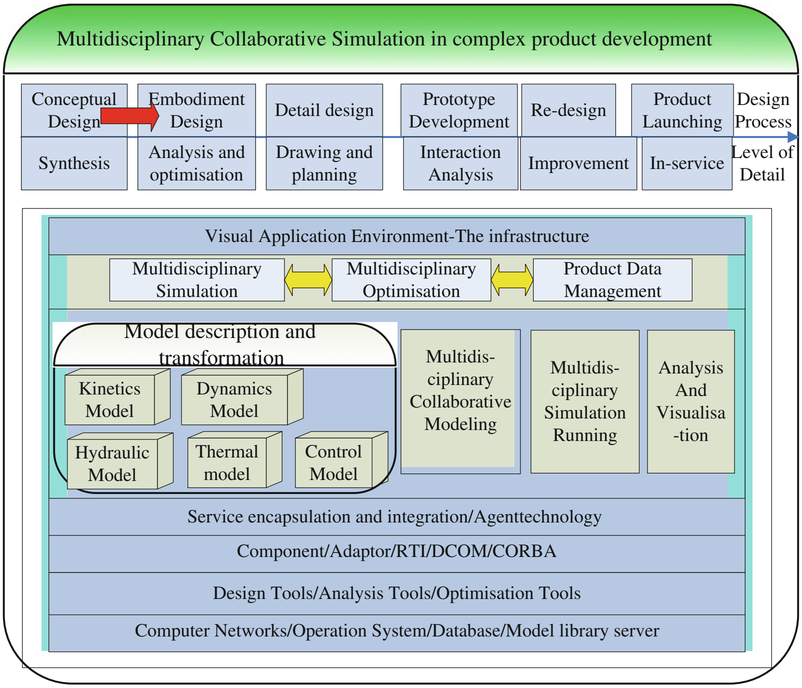

Fig. 8.3 A framework for multidisciplinary collaborative simulation systems

constructed using professional CAE packages. A prerequisite for multidisciplinary collaborative simulation is that distributed models should have uniform interfaces, and thus requires the development of a mechanism for model description and transformation. The collaborative modeling tool is based on such a mechanism and aims to hide technical details about both the implementation of, and the interaction between, simulation models. At simulation run-time, a tool that can monitor and control the process of simulation is also necessary. Last but not least, tools for data analysis and visualization are also helpful to evaluate the simulation effectively and efficiently. In the infrastructure tier, various systems, platforms and tools are operating and communicating on the information infrastructure in enterprises. Disciplinary models and optimization algorithm are implemented using specific CAE tools which are usually heterogeneous and have no interfaces between each other. As a consequence, middleware technology, component technology, and software adaptors should be utilized to facilitate the access to these models. Furthermore, open standards such as Web service are helpful for supporting interoperability in a simulation involving distributed heterogeneous models. Based on these techniques, the infrastructure can facilitate the creation of disciplinary models, enable the interactions between them, and provide solutions for underpinning collaborative simulation running.

8.5.3 Prototype System Implementation

To evaluate the service-oriented solution, a prototype system has been developed. Currently, the system has functionalities of collaborative modeling, simulation running, and simple post-processing. The services implemented include information services, simulation services, and integration services. Collaboration services have also been developed to support collaborative work in a distributed virtual environment, but core functions such as synchronous operation and complex data exchange, nevertheless, have not been implemented. The distributed computing technologies used for this solution are Web Services and the High Level Architecture (HLA). The prototype system is developed as a Web-based platform which can be assessed by engineers and analysts wherever they are based to create models, run simulation, and analyze simulation results. For the sake of brevity, this book chapter mainly describes the integration of Web Services and HLA in this solution. Detailed discussion on the technologies used in this solution can be found in [41], and readers interested in more details about the system are referred to other relevant publications [7].

The combinative use of Web Services and the HLA is taken as the solution underpinning the prototype system. The solution is illustrated in Fig. 8.4 where MVMD stands for multi-view model description. Specifically, the HLA-based federation is used to support run-time data exchange, which runs in Local Area Network (LAN) to achieve improved efficiency. The simulation of individual models is executed by performing numerical integration using the solvers embedded in the commercial CAE tools which are used to create these models. To implement the distributed computation and interaction of these models, each model is encapsulated as a Web service which is accessible to remote service subscriber. This enables the simulation functionality (encapsulated as Web Services) and run-time interaction (implemented in HLA federation) to be separated

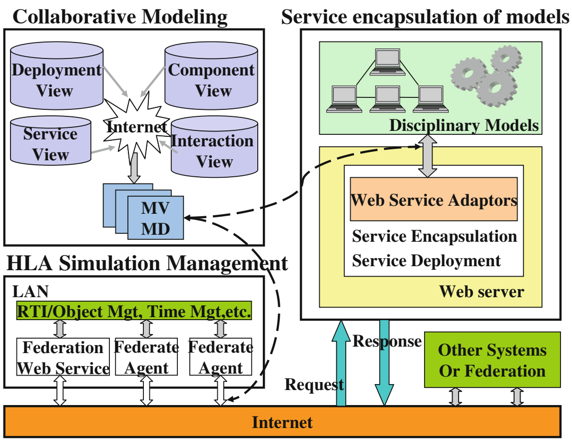

Fig. 8.4 A solution based on web services and HLA

as independent modules, which implements the flexible integration of a simulation system, i.e. changes in any module can be addressed locally. Thereby, the distributed and interactive features are all supported by the run-time integration of HLA and Web Services. Moreover, a multidisciplinary modeling paradigm is introduced to provide the users with a high-level tool for describing a simulation system. The information obtained in the multidisciplinary modeling is used to guide the construction of HLA federation and Web Services encapsulation and development. Some snapshots of the Graphical User Interfaces (GUIs) are shown in Fig. 8.5. It is noteworthy that this is still a preliminary stage of implementing a collaborative simulation platform and further work is still needed to improve the system and address performance issues.

8.5.4 The Design Process and a Simulation Example

To illustrate the performance of the prototype system, the design process of multidisciplinary simulations is analyzed and a simulation example is run on it. Figure 8.6 shows the six-stage process together with the developers and services involved at different stages.

The prototype system can assist users (design engineers, project managers, IT engineers, and system analysts) to perform tasks at these stages by providing Web-based interfaces and integrating relevant services. Specifically, the first stage is the analysis of simulation requirements, which involves the participation of design engineers, project managers, and system analysts to identify requirements for the simulation problem with the support of information services (e.g. specification, available models, and information about previous projects). The synthesis stage heavily relies on the experience and knowledge of design engineers and system analysts, and thus accessing to information/knowledge services is quite helpful. The development of simulation models is mainly completed by design engineers with the support of design services and involves two levels of modeling work. The first level is system modeling at which design engineers work together to decompose a complex system as subsystems. The subsequent subsystem modeling is the second level which involves the creation of subsystem models for different disciplines.

The subsystem models are mainly created using either commercial CAE tools or bespoke packages. IT engineers and system analysts work together to make these models assessable on the Internet at the next stage namely encapsulation of models as services. In this way, these models can be integrated at simulation run-time. Therefore, simulation and integration services are needed at this stage. Once these models are made assessable, collaborative simulation can be started which enables the collective use of simulation knowledge and computational resources. The running stage is the core of the design process and therefore involves the participation of engineers, analysts, and IT experts. The last stage is post-processing and system analysis, which involves the analysis of simulation results

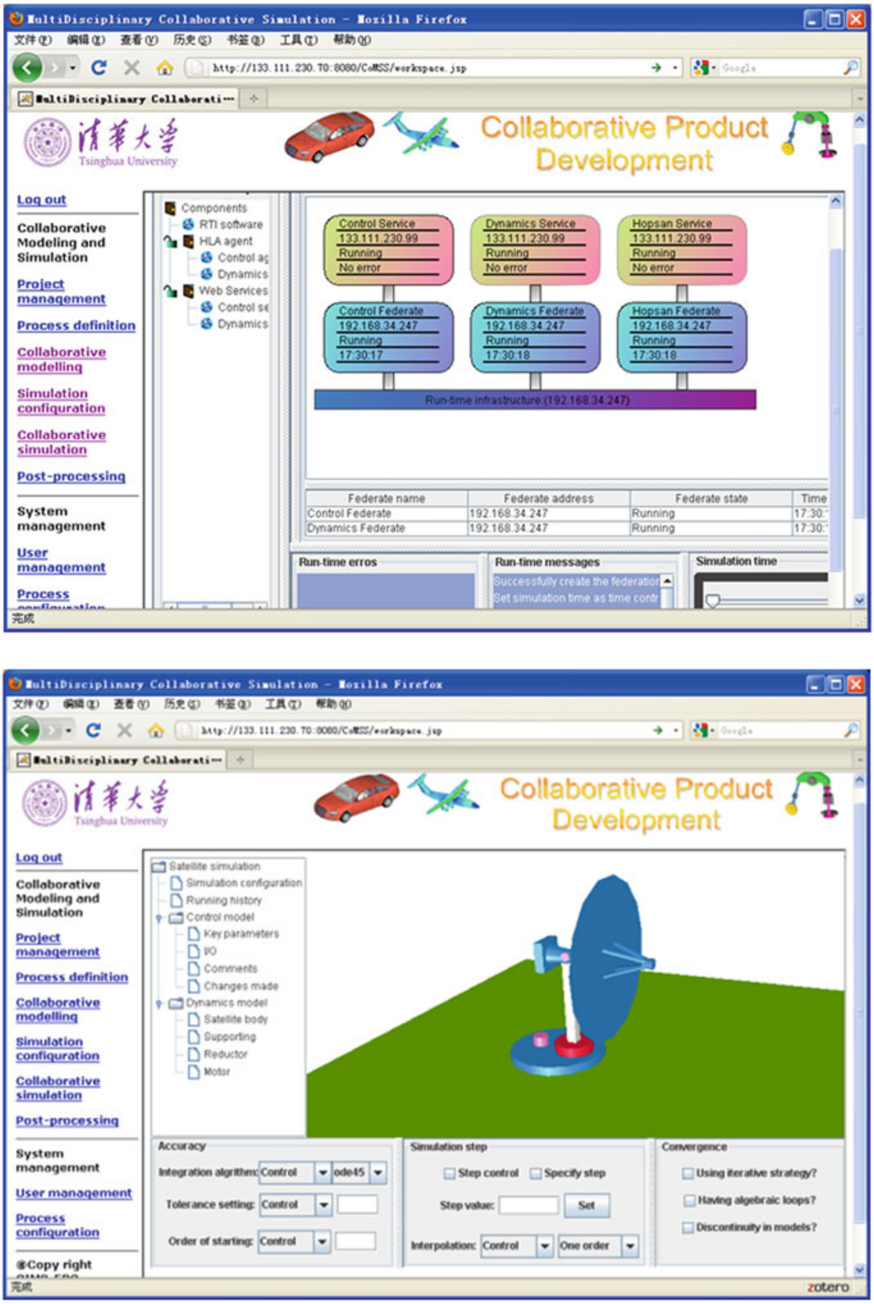

Fig. 8.5 Snapshots of the GUI of the prototype system

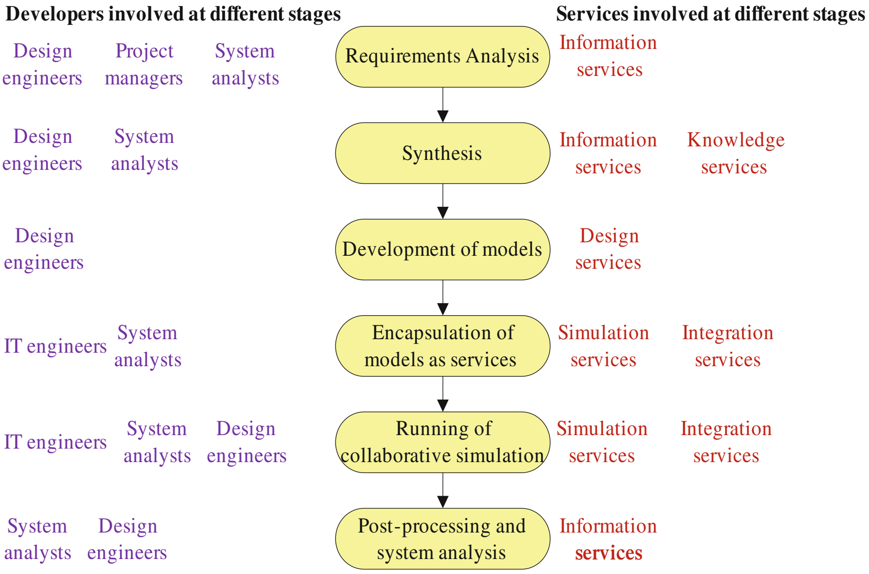

Fig. 8.6 The design process based on the prototype system

and the identification of areas for improvement and optimization. At this stage, design engineers and system analysts evaluate the performance of a design solution and record relevant information for later usage.

The simulation of the tilting process of a satellite is run on the prototype system. This example is shown in Fig. 8.5, which involves two models, namely a control model and a dynamics model. The objective of simulation is to evaluate whether the control algorithm can successfully drive the satellite body to tilt for a specified angle. Engineers work on the prototype system step by step and the developed models are encapsulated as Web services. At the running stage, users can control the simulation process and retrieve simulation results even though none of the CAE packages is installed in their computers. The current prototype can support simple post-processing and the simulation results for the satellite example are shown in Fig. 8.7. The total simulation is 0.25 s and the interval for data exchange is set as 0.005 s. This means that the simulation involves 50 data exchanges and the time taken for each exchange is about 1 s. It is noteworthy the initialization of simulation engines takes a relatively long time and in practice they should be initialized before the simulation starts. In summary, the prototype system can assist users to create models, run simulations, and undertaken system analysis by integrating specialized services. Therefore, it is possible to support design tasks using DBS. Issues (e.g. identification of services, use of computing technologies, and the integration of information and computing resources)

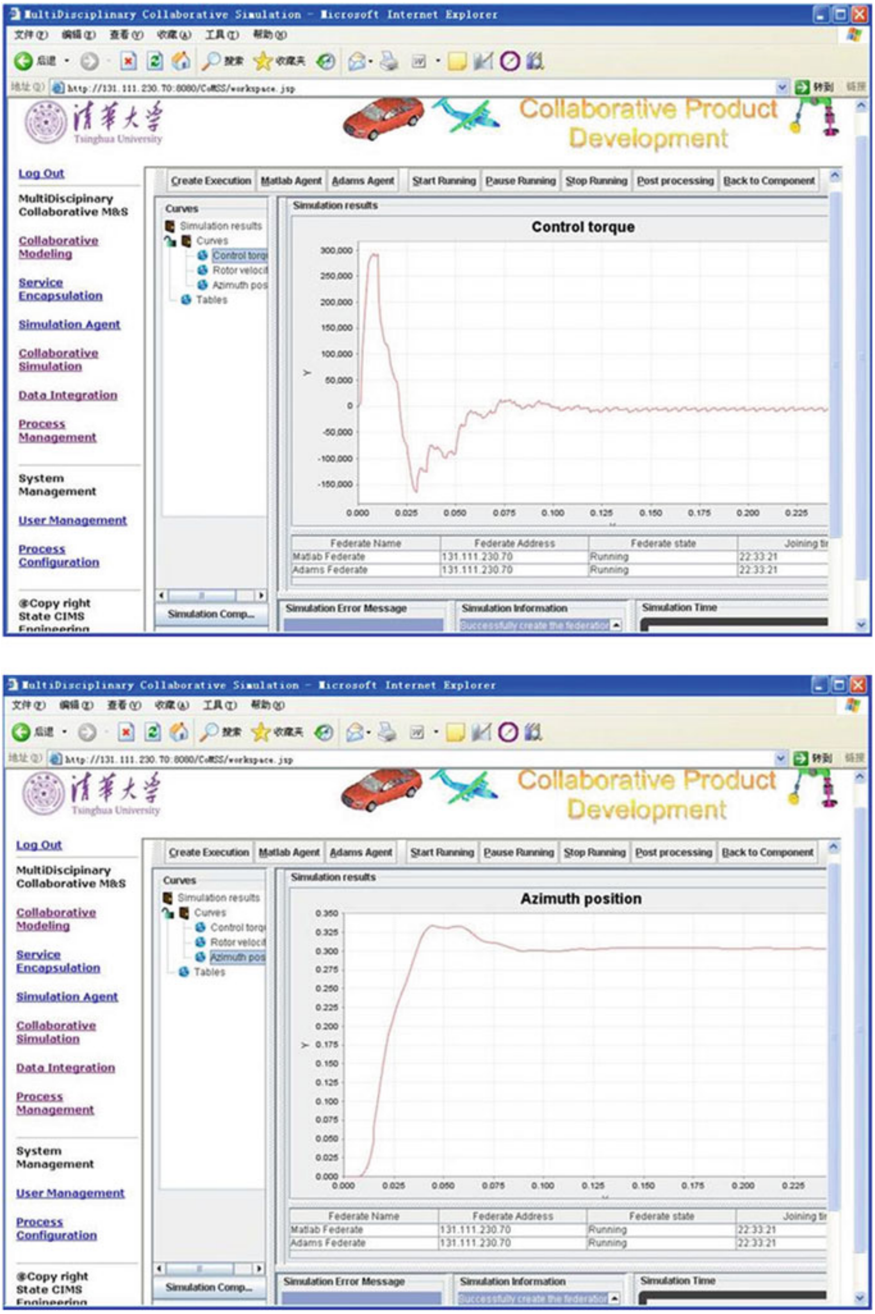

Fig. 8.7 Simulation results shown on the GUI of the prototype system

addressed in the development of this prototype can also be taken into account in the development of more powerful DBS systems. This suggests that DBS is implementable and promising for supporting complex product development.

8.6 Conclusion Remarks

This book chapter presents a new scheme for CPD, namely designing by services, which is aimed at supporting effective and flexible system integration by supplying professional, accessible, reliable, extensible, and secure services. Information, collaboration, integration, and computation are identified as the key issues for CPD, and the proposed scheme emphasizes the provision of tasks concerning these issues as services. In this way, design tasks can be underpinned by an information infrastructure which operates by accessing and integrating relevant services. This scheme has a few advantages which are difficult to achieve for traditional methods. Firstly, services are provided by teams who have the specialized information/resources and consumers of these services only need to know the service interfaces rather than the technical details. This helps achieve effective and efficient division of tasks to facilitate carrying out design projects in a collaborative way. Secondly, changes incurred in any service component can be to a large extent addressed locally as system integration is mainly done through interfaces. Thirdly, distributed collaboration is enabled with the support of modern distributed computing technologies.

Eight kinds of services have been identified in a service-oriented architecture for the proposed scheme. Their purposes together with their usage in a design scenario are described. These services are sufficient for general CPD processes and in practice they can be evaluated and selected based on specific requirements. The key enabling technologies for designing by services are identified and discussed in detail. These technologies are very important for the successful implementation of design systems for the proposed scheme. The design and implementation of a multidisciplinary collaborative simulation system is described as a case study. Preliminary work on the development of the prototype shows that the issues raised in DBS can guide the development of solutions for collaborative simulation and the use of advanced computing technologies can greatly help system implementation. DBS is a promising yet complicated topic which requires much further work. The preliminary work presented in this chapter can be used as reference models for the implementation of the next-generation distributed and collaborative engineering software applications. Our future work includes improving and evaluating the prototype system, as well as producing detailed solutions for different services in the proposed scheme.

Acknowledgments The authors acknowledge the support from the National Natural Science Foundation of China (Grant No. 61074110).

References

1. Wallace K (2011) Transferring design methods into practice. In: Birkhofer H (ed) The future of design methodology. Springer, London, pp 239–248
2. Szykman S, Sriram RD (2006) Design and implementation of the Web-enabled NIST design repository. ACM Trans Internet Technol 6:85–116
3. Szykman S, Fenves SJ, Keirouz W, Shooter SB (2001) A foundation for interoperability in next-generation product development systems. Comput Aided Des 33:545–559
4. Rodriguez K, Al-Ashaab A (2005) Knowledge web-based system architecture for collaborative product development. Comput Ind 56:125–140
5. Wang HW, Zhang HM (2008) An integrated and collaborative approach for complex product development in distributed heterogeneous environment. Int J Prod Res 46:2345–2361
6. Abrahamson S, Wallace D, Senin N, Sferro P (2000) Integrated design in a service marketplace. Comput Aided Des 32:97–107
7. Wang H, Zhang H (2010) A distributed and interactive system to integrated design and simulation for collaborative product development. Robot Comput-Integr Manuf 26:778–789
8. Li WD, Lu WF, Fuh JYH, Wong YS (2005) Collaborative computer-aided design—research and development status. Comput Aided Des 37:931–940
9. Shen W, Hao Q, Mak H, Neelamkavil J, Xie H, Dickinson J, Thomas R, Pardasani A, Xue H (2010) Systems integration and collaboration in architecture, engineering, construction, and facilities management: a review. Adv Eng Inform 24:196–207
10. Shen W, Hao Q, Li W (2008) Computer supported collaborative design: retrospective and perspective. Comput Ind 59:855–862
11. Shyamsundar N, Gadh R (2002) Collaborative virtual prototyping of product assemblies over the Internet. Comput Aided Des 34:755–768
12. Kim K-Y, Manley DG, Yang H (2006) Ontology-based assembly design and information sharing for collaborative product development. Comput Aided Des 38:1233–1250
13. Wang JX, Tang MX, Song LN, Jiang SQ (2009) Design and implementation of an agent-based collaborative product design system. Comput Ind 60:520–535
14. Li WD, Ong SK, Nee AYC (2005) A Web-based process planning optimization system for distributed design. Comput Aided Des 37:921–930
15. Lee H, Kim J, Banerjee A (2010) Collaborative intelligent CAD framework incorporating design history tracking algorithm. Comput Aided Des 42:1125–1142
16. Park J-H, Seo K-K (2006) A knowledge-based approximate life cycle assessment system for evaluating environmental impacts of product design alternatives in a collaborative design environment. Adv Eng Inform 20:147–154
17. Curran R, Gomis G, Castagne S, Butterfield J, Edgar T, Higgins C, McKeever C (2007) Integrated digital design for manufacture for reduced life cycle cost. Int J Prod Econ 109:27–40
18. Rahmani K, Thomson V (2012) Ontology based interface design and control methodology for collaborative product development. Comput Aided Des 44:432–444
19. Wang K, Takahashi A (2012) Semantic Web based innovative design knowledge modeling for collaborative design. Expert Syst Appl 39:5616–5624
20. Jiang P, Shao X, Qiu H, Gao L, Li P (2009) A web services and process-view combined approach for process management of collaborative product development. Comput Ind 60:416–427
21. Shen W, Hao Q, Wang S, Li Y, Ghenniwa H (2007) An agent-based service-oriented integration architecture for collaborative intelligent manufacturing. Robot Comput-Integr Manuf 23:315–325
22. Liu D, Peng J, Law KH, Wiederhold G, Sriram RD (2005) Composition of engineering web services with distributed data-flows and computations. Adv Eng Inform 19:25–42

23. Pullen JM, Brunton R, Brutzman D, Drake D, Hieb M, Morse KL, Tolk A (2005) Using web services to integrate heterogeneous simulations in a grid environment. Future Gener Comput Syst 21:97–106
24. Szykman S, Racz J, Bochenek C, Sriram RD (2000) A web-based system for design artifact modeling. Des Stud 21:145–165
25. Chu C-H, Wu P-H, Hsu Y-C (2009) Multi-agent collaborative 3D design with geometric model at different levels of detail. Robot Comput-Integr Manuf 25:334–347
26. Shen Q, Grafe M (2007) To support multidisciplinary communication in VR-based virtual prototyping of mechatronic systems. Adv Eng Inform 21:201–209
27. Zha XF, Du H (2006) Knowledge-intensive collaborative design modeling and support: part I: review, distributed models and framework. Comput Ind 57:39–55
28. Robin V, Rose B, Girard P (2007) Modelling collaborative knowledge to support engineering design project manager. Comput Ind 58:188–198
29. Wang H, Johnson A, Zhang H, Liang S (2010) Towards a collaborative modeling and simulation platform on the Internet. Adv Eng Inform 24:208–218
30. Shephard MS, Beall MW, O'Bara RM, Webster BE (2004) Toward simulation-based design. Finite Elem Anal Des 40:1575–1598
31. Arnold M, Heckmann A (2007) From multibody dynamics to multidisciplinary applications. In: García Orden JC, Goicolea JM, Cuadrado J (eds) Multibody dynamics, vol 4. Springer, Netherlands, pp 273–294
32. Samin J, Brüls O, Collard J, Sass L, Fisette P (2007) Multiphysics modeling and optimization of mechatronic multibody systems. Multibody Sys Dyn 18:345–373
33. Kübler R, Schiehlen W (2000) Modular Simulation in multibody system dynamics. Multibody Sys Dyn 4:107–127
34. Xiang W, Fok SC, Thimm G (2004) Agent-based composable simulation for virtual prototyping of fluid power system. Comput Ind 54:237–251
35. Roselló EG, Lado MJ, Méndez AJ, Dacosta JG, Cota MP (2007) A component framework for reusing a proprietary computer-aided engineering environment. Adv Eng Softw 38:256–266
36. Tsai WT, Fan C, Chen Y, Paul R (2006) A service-oriented modeling and simulation framework for rapid development of distributed applications. Simul Model Pract Theory 14:725–739
37. Wang YD, Shen W, Ghenniwa H (2003) WebBlow: a Web/agent-based multidisciplinary design optimization environment. Comput Ind 52:17–28
38. Cheng H-C, Fen C-S (2006) A web-based distributed problem-solving environment for engineering applications. Adv Eng Softw 37:112–128
39. Fan LQ, Senthil Kumar A, Jagdish BN, Bok SH (2008) Development of a distributed collaborative design framework within peer-to-peer environment. Comput Aided Des 40:891–904
40. Reed JA, Follen GJ, Afjeh AA (2000) Improving the aircraft design process using web-based modeling and simulation. ACM Trans Model Comput Simul 10:58–83
41. Byrne J, Heavey C, Byrne PJ (2010) A review of web-based simulation and supporting tools. Simul Model Pract Theory 18:253–276
42. Wang H, Zhang H, Johnson A (2009) A service-oriented approach for the collaborative simulation of complex engineering systems. In: Proceedings of 2009 world conference on services–I, pp 78–84
43. Wang H, Zhang H (2006) Collaborative simulation environment based on HLA and web service. In: Proceedings of 10th international conference on computer supported cooperative work in design, CSCWD'06, pp 1–6
44. Pahl G, Beitz W, Wallace K, Blessing L (2007) Engineering design: a systematic approach. Springer, London
45. Ahmed S, Wallace KM (2004) Understanding the knowledge needs of novice designers in the aerospace industry. Des Stud 25:155–173

46. Wang H, Johnson AL, Bracewell RH (2012) The retrieval of structured design rationale for the re-use of design knowledge with an integrated representation. Adv Eng Inform 26:251–266
47. Manning CD, Raghavan P, Schutze H (2008) Introduction to information retrieval. Cambridge University Press, Cambridge, MA
48. Aurisicchio M, Bracewell R, Wallace K (2009) Understanding how the information requests of aerospace engineering designers influence information-seeking behaviour. J Eng Des 21:707–730