Moving Integrated Product Development to Service Clouds in the Global Economy
J. Cha et al. (Eds.)
© 2014 The Authors and IOS Press.
This article is published online with Open Access by IOS Press and distributed under the terms of the Creative Commons Attribution Non-Commercial License.
doi:10.3233/978-1-61499-440-4-278

Research on Software Resource Sharing Management in Collaborative Design Environment Based on Remote Virtual Desktop

Wensheng XU[a,1], Nan LI[b], Hong TANG[a], Jianzhong CHA[a]
[a] *School of Mechanical, Electronic and Control Engineering, Beijing Jiaotong University, Beijing, China*
[b] *School of Material and Mechanical Engineering, Beijing Technology and Business University, Beijing, China*

Abstract. For complex product development, different experts from different domains may use various engineering software tools to collaborate in the product development process in a distributed environment. An engineering software resource pool needs to be built to facilitate efficient software resource sharing and management in the collaborative design process. Engineering software tools can be encapsulated either as SOA services through SOA based technologies, or as interactive desktop services through remote virtual desktop technology according to their different interfaces and requirements. For interactive software resources, a software resource sharing conceptual model is analyzed in this paper, and a software resource sharing framework is proposed for collaborative design based on the remote virtual desktop technology. Based on Ulteo – an open-sourced remote virtual desktop platform, a software resource sharing platform is designed and implemented for high-speed train axle lightweight design, in which relevant software tools are managed and provisioned as remote desktop interactive software services and can be shared and accessed in a distributed environment. Through the software resource sharing platform, users can customize their work desktop with different software resources according to their own demands, and various resource sharing modes can be well supported.

Keywords. Collaborative Design, Software Resource Sharing, Interactive Software, Remote Virtual Desktop

Introduction

For complex product development, design tasks and multi-functional teams may be organized according to the product structural tree, and in each multi-functional team different experts from different domains may use various engineering software tools to collaborate in the product development process in a distributed environment. A large amount of engineering software tools may be involved in the development process, for example, a 3-D modelling tool – Pro/E, a meshing tool – HyperMesh, a finite element analysis tool – ANSYS, etc. An engineering software resource pool needs to be built to

[1] Corresponding Author: wshxu@bjtu.edu.cn

facilitate efficient software resource sharing and management in the collaborative design process, so developers from distributed locations can easily and conveniently access and use those engineering tools and transfer intermediate data among team members. With the support of an engineering software resource pool, there is no need to install engineering software tools locally for each developer in the product development process, and the utilizing efficiency of all the software tools can be well enhanced. A resource cloud pool is the basis for the implementation of cloud manufacturing [1,2], and it can effectively enhance the utilizing efficiency of manufacturing software resources. In the resource cloud pool, resources should be encapsulated as discoverable, usable services through servicing technology or virtual technology [3,4], so the heterogeneous structures of the resources and their underlying environment can be hidden from end users, resources can be well shared in the cloud manufacturing environment.

Software resources are located at the bottom level of the whole cloud manufacturing architecture, and they need to be encapsulated and shared according to their different interfaces and user requirements. There are basically 3 types of software interfaces for engineering software tools: command line interface, API interface, and interactive user interface. Software interfaces and user requirements will determine how a software resource can be encapsulated, shared and presented in the design process. For command line interfaces and API interfaces, the software resources can be encapsulated as SOA-based services, then other applications can conveniently invoke these services as needed, as shown in our previous work [5,6]. For interactive user interfaces, software resources can be encapsulated and presented in the form of remote virtual desktop, so the complete features of the software tools can be presented for users. Users can interact with these software services remotely as if the applications are located locally.

Currently there are 2 types of virtual desktop solutions: Virtual Desktop Infrastructure (VDI) and Server-based Computing (SBC) (also called remote desktop services or presentation virtualization) [7]. A VDI solution provides full desktops for remote end users, each user with an individual operating system instance. SBC sessions run in a single shared server operating system and can provide connections to either individual applications or the full desktop as needed. Now there are two main commercial software solutions for virtual desktops in the market: VMware products and Citrix products. VMware mainly provides the VDI virtual desktop solution, focusing on enterprise level applications based on data centers. Citrix XenDesktop is an SBC virtual desktop solution and focusing on usage on virtualization on terminals [8]. These two types of virtual products have limited open further development API, so small and medium-sized enterprises have difficulties in further development and customization according to their different requirements and resource share modes; furthermore, the prices of their products are too costly for small and medium-sized enterprises.

To implement interactive software resource sharing and on-demand services in cloud environment in an affordable and flexible way for small and medium-sized enterprises, a resource sharing concept model is proposed based on virtual technology in this paper, and a software resource sharing framework based on virtual technology is proposed. An open-sourced virtual desktop platform – Ulteo [9] is adopted to construct and develop the software resource sharing platform, so interactive software services are encapsulated and presented with virtual technology. All functions of the software resources can be presented to users, and software resources either in Linux or in

Windows environment can be easily encapsulated, shared and integrated in the platform, and services can be provided for small and medium-sized enterprises in an on-demand fashion.

1. The conceptual model of software resource sharing

Software resources are indispensable resources in enterprise business processes, and they are distributed in different enterprises and different regions, or in different departments in an enterprise, and they have some unique characteristics, including being able to be virtualized, copied and shared. A specific software resource can be denoted as sr_i, a software resource set can be denoted as $SR=\{sr_1, sr_2, sr_3, \ldots, sr_n\}$. Ju et al. proposes a software resource sharing model based on distributed network model, composed of location set, software resource domain set, software location set, software resource set and sharing relations set [10], but in that model the relations between the software resources and the users are not clearly defined and cannot clearly describe all possible sharing modes for users. To effectively and clearly express the sharing relations of software resources, a software resource sharing conceptual model (SRSCM) is proposed in this paper, as shown in Figure 1. A SRSCM is a quintuple set -- SRSCM={P, SR, SRZ, S, UV}, in which P, SR, SRZ, S and UV refer to location set, software resource set, sharing resource zone, sharing relation set and user view set respectively:

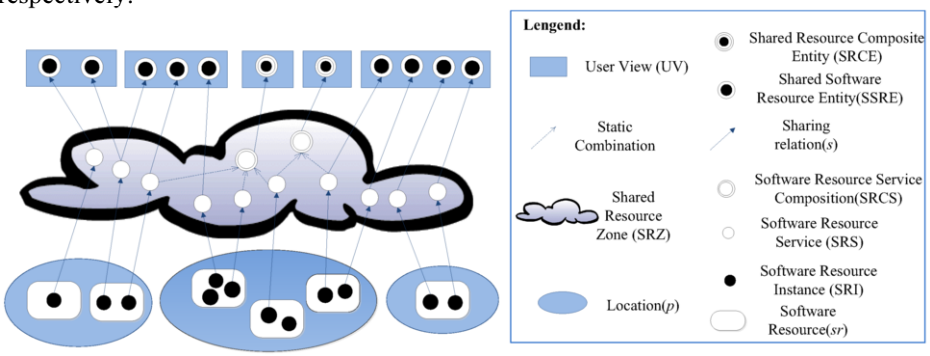

Figure 1. The conceptual model of software resource sharing.

(1) A location set includes the locations of software resources in an enterprise or in a department. Let p_i denote a location, then a location set $P=\{p_1, p_2, \ldots, p_n\}$.

(2) A software resource set is the set of software resources needed in manufacturing activities. Let sr_i denotes a software resource of some type, then software resource set $SR=\{sr_1, sr_2, sr_3, \ldots, sr_n\}$.

Each sr_i is located in some location p_j and can have one or several software resource instances (SRI), and each SRI is a running instance of the software resource.

(3) Shared Resource Zone (SRZ) is a set of resource services which can provide services for users. It includes a number of software resource service (SRS) and software resource composite service (SRCS). Each SRS is formed through encapsulating a corresponding SRI as a service. SRCS is the result of static integration of multiple SRS.

(4) User view (UV) is the current view of the available shared resources for end users, it is composed of one or more shared software resource entities (SSRE) and/or

shared software resource composite entities (SRCE). It can be presented in the form of virtual desktop. Users can access the SSRE or SRCE through the virtual desktop. Each SSRE is formed through sharing an SRS, while an SRCE is formed through sharing an SRCS.

(5) Sharing relations are composed of two types of links – the directed service encapsulation links from an SRI to some SRS, denoted as se_i, and the directed software presentation links from an SRS to some SSRE or from an SRCS to some SRCE in a user view, denoted as sp_i. Then sharing relation set $S=\{ se_1, se_2, \ldots, se_n, sp_1, sp_2, \ldots, sp_m \}$.

Various software resources can be encapsulated as resource services through virtual technology, all the software resource services form a software resource cloud pool by central scheduling and management, and on-demand sharing and service can be implemented. Based on the above concepts, all possible sharing modes of software resources can be expressed in enterprises. For example, multiple SRS can be utilized by one user view, i.e. a user view can simultaneously utilize multiple software resource services in a unified local environment, and it forms a "multiple services to one user" relation; or an SRS can be shared by two user views, so it forms a "one service to multiple users" relation, as shown in Figure 2.

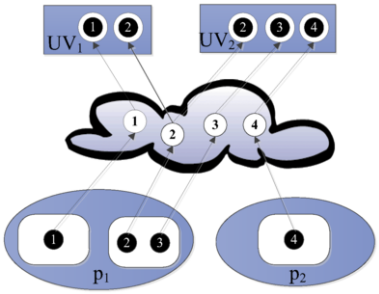

Figure 2. Different sharing modes between software services and users.

2. Software resource sharing framework based on virtual desktop

Based on the conceptual model of software resource sharing, in order to better utilize distributed heterogeneous software resources and implement effective and on-demand services, a software resource sharing framework based on virtual desktop is proposed. The software resource sharing framework can be divided into four layers: software resource layer, virtualization platform layer, virtual resource cloud pool layer, and application layer, as shown in Figure 3. The four layers work consistently, so distributed heterogeneous software resources can be shared, and it can provide on-demand services for users, and users can access these software resources transparently without knowing their underlying environments.

The application layer is located at the top of the whole architecture, and it provides applications for end users, and interacts with users. So application layer can also be called human-machine interaction layer. It provides on-demand user interface in the form of remote virtual desktop. Different user groups may have different applications. We can use user groups to manage the user roles, so as to manage the sharing requirements of different users in groups. Virtual software resource services form the virtual resource pool. Users can search, select and combine needed software services, and software resources can hide their distributed and heterogeneous features for users.

On-demand and personalized resource service providing can be achieved by service selections by users. The virtualization layer provides virtual desktop technology to support desktop virtualization. Desktop virtualization is software technology that separates the desktop environment and associated application software from the physical client device that is used to access it [7]. Interactive software resources are wrapped and virtualized as services through virtual desktop technology, and they can be managed centrally and shared by distributed users. Various software resources located in different distributed places form the software resource layer. The software resource layer is the actual service providers for end users, and it is the running basis for the whole framework.

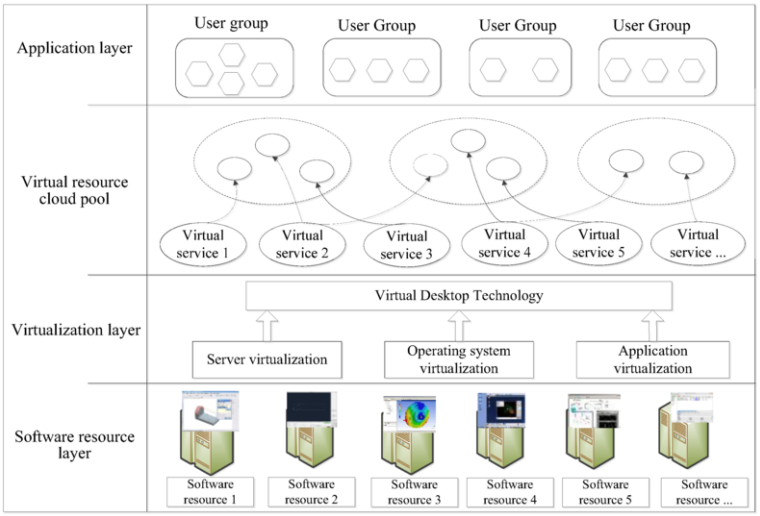

Figure 3. Software resource sharing framework based on virtual desktop.

3. Implementation of the software resource sharing platform

3.1. Structure of the software resource sharing platform

According to the framework, based on the Ulteo virtual desktop technology, the structure of the software resource sharing platform is shown in Figure 4. The virtual desktop platform – Ulteo is used to implement the interactive software resource sharing platform. Ulteo -- Open Source Enterprise Virtual Desktop and Application Delivery solutions (Open Source VDI & SBC), is an open-sourced virtual desktop, and it can be tailored and further developed according to users' requirements. Ulteo is based on Debian and Ubuntu OS, and it can allow users to run any applications on Linux and Microsoft Windows through Web browsers.

The session manager server is the core for software resource sharing. Inside the session manager server, there are four management modules: user management module, service host management module, virtual service management module, system management module. Application servers in which engineering software tools are running can register with the service host management module, and then the virtual service management module can maintain an interactive software service list in

accordance with the application servers. The session manager server also provides 2 management portals: platform management portal and user self-management portal. Through the platform management portal, administrators of the resource sharing platform can manage the platform, including shared software joining, suspending, removing, etc. Through the user self-management portal, end users can register with this platform, and search and select interactive software tools in the platform, so on-demand software services can be achieved. The platform system management module is responsible for monitoring the platform, including status monitoring, log management, system performance measurement, billing management. Based on this platform, software resources can be rent to users, users can apply for software resources according to their own needs and also financial constraints if billing is applied. A user who works on the Web Client (a desktop or laptop computer) can use a web browser to access the web portal server and then login there and select the needed software resources to use.

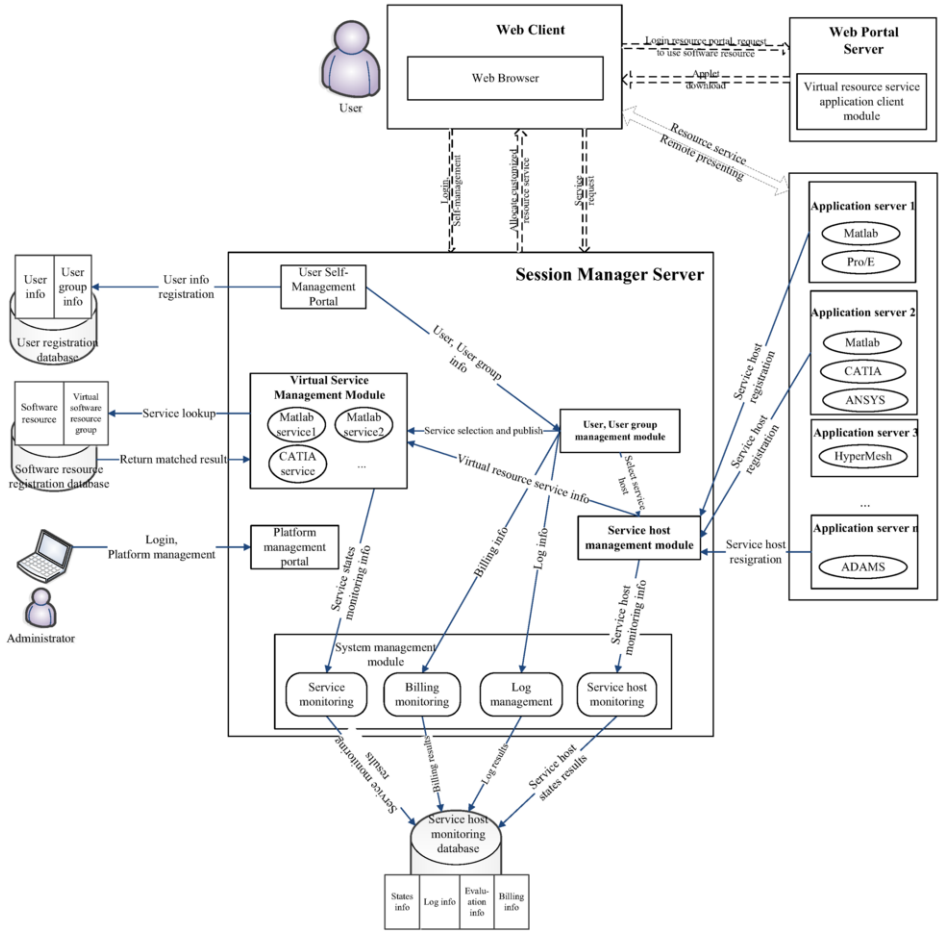

Figure 4. Structure of the software resource sharing platform.

3.2. Communication processes in the software resource sharing platform

In the software resource sharing platform based on Ulteo, when a web client accesses the platform web portal through the HTTP protocol, after login, a Remote Display Protocol (RDP) Java applet which is developed with properJavaRDP [11] is downloaded from the portal and will run in the web browser on the web client. Subsequently a session will be established between the web client and the session manager and a token will identify the session. With the provided information from the session manager, the RDP Java applet will then try to establish an application connection between the web client and the intended application server based on the RDP protocol on top of the SSL secure layer. After SSL handshaking, client-server authentication, data communication channel establishment and graphical data connection verification, then the web client can receive the RDP data packages from the application server and present a remote virtual desktop for the user. The user can then interact with the applied software tools on the application servers remotely through the remote virtual desktop. The communication processes in the platform based on Ulteo are shown in Figure 5.

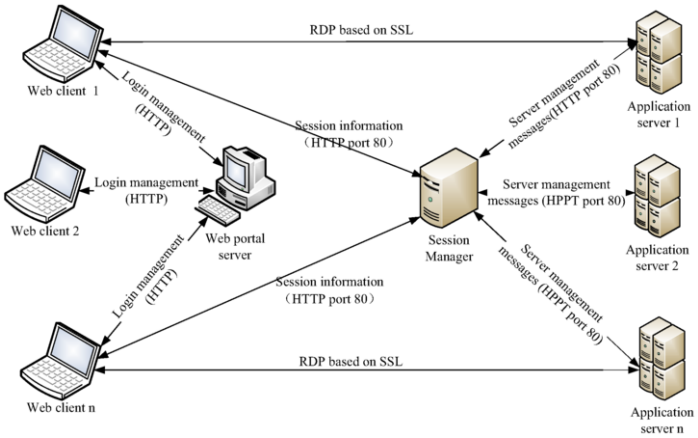

Figure 5. Communication processes in the software resource sharing platform.

3.3. Example of the software resource sharing for lightweight design of axles

To take the lightweight design of high speed train axles for example, the development process and the needed engineering software tools are shown in Figure 6. Once a software resource pool is set up including all the needed engineering software tools, developers can register at the platform, and select needed software services in the platform. The Pro/E, HyperMesh and ANSYS software tools run in Windows environment, while the MATLAB tool is running in Linux environment. Developers can access the software services remotely in the virtual desktop, and interact with the software tools as needed, without needing to know the underlying running environment of the software tools, as shown in Figure 7. For Developer 3 from Figure 7 in the axle lightweight design project, the remote desktop is shown in Figure 8, in which the MATLAB in Linux environment and the ANSYS in Windows environment can be accessed through the web browser in the local environment.

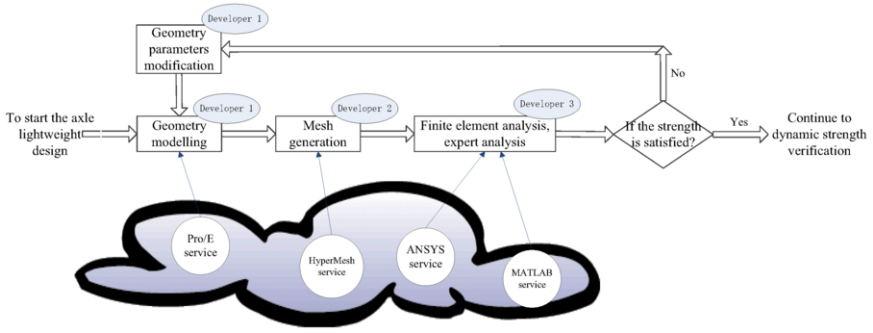

Figure 6. Process of lightweight design of axles.

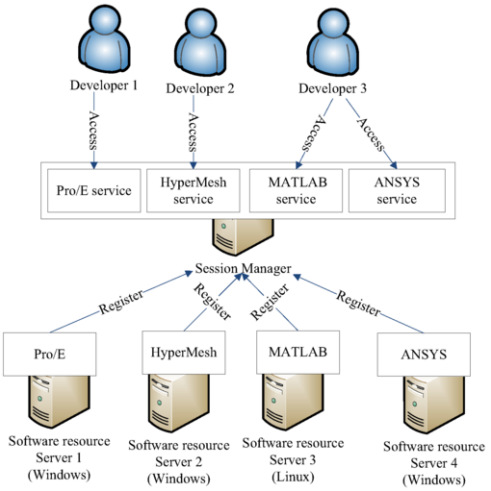

Figure 7. Developers and their needed software resources.

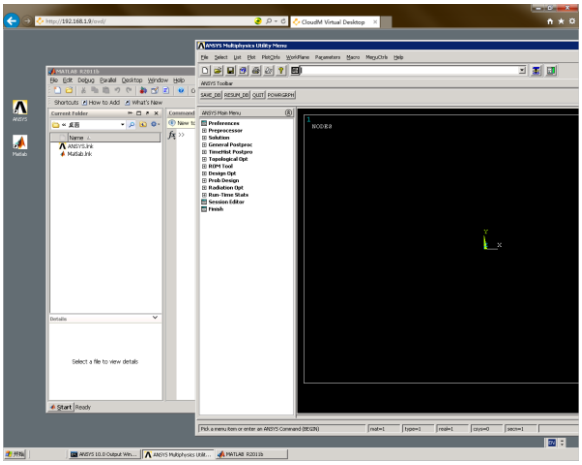

Figure 8. The work desktop for Developer 3 with the MATLAB and ANSYS software service.

4. Conclusions

Software resources can be encapsulated and provisioned as remote desktop interactive software services and can be shared and accessed in a distributed environment. The underlying distributed features and heterogeneous environment of the software resources can be hidden from the end users through virtual desktop technology. Users can select their needed software resources, either from Linux environment or from Windows environment, in an on-demand fashion, and can interact with the software tools in the remote virtual desktop in a unifying environment through a web browser. With the virtual desktop sharing framework, interactive software resources can be encapsulated and shared conveniently and effectively within and among enterprises, and the product development process can be well supported.

Acknowledgement

This work is supported by the National Natural Science Foundation of China (51175033) and National High Technology Research and Development Program of China (2013AA041302).

References

[1] Chunquan Li, Chunyang Hu, Yanwei Wang, Research of resource virtualization technology based on cloud manufacturing, *Advanced Materials Research*, vol.201-203 (2001), 681- 684.
[2] Lin Zhang, Yongliang Luo, Fei Tao, et al, Study on the key technologies for the construction of manufacturing cloud, *Chinese Journal of Computer Integrated Manufacturing Systems*, 16(11) (2010), 2510-2520 (in Chinese).
[3] Lei Wu, Xiangxu Meng, Shijun Liu, Service-oriented encapsulation of manufacturing resources, *Proceedings of International Conference on Services Computing*, July, 2007, Salt Lake City, USA: SCC, pp.727- 728, 2007.
[4] Jiri Vorisek, Business Drivers for Application Servicing and a Software-as-a-Service Model. *Proceedings of the Fourth International Conference on Electronic Business*, 2004, Beijing, pp. 511-516.
[5] L.J. Kong, W.S. Xu, N. Li, J.Z. Cha, Research on service encapsulation of manufacturing resources based on SOOA. *Advances in Information Sciences and Service Sciences*, 5(1) (2013) 158-166.
[6] Wensheng Xu, Lingjun Kong, Nan Li, Jianzhong Cha, A Service Encapsulation Method in Cloud Simulation Platform, *Communications in Computer and Information Science*, 10 (2012) 431-439.
[7] http://en.wikipedia.org/wiki/Desktop_virtualization
[8] http://www.citrix.com/products/xenapp/overview.html
[9] http://www.ulteo.com/home
[10] Wenjun Ju, Linfu Sun, Huijuan Zhao, et al. Research on software resource sharing based on server-based computing, *Chinese Journal of Computer Integrated Manufacturing Systems*, 11(10) (2005) 1486-1490(in Chinese).
[11] http://properjavardp.sourceforge.net/