Title:
Service Oriented Cross-platform Framework for 3D Part Library Based on Diversiform Knowledge Implantation

Authors:
Wei Liu, liu-wei@sjtu.edu.cn, Shanghai Jiao Tong University
Qiang Niu, qniu@sjtu.edu.cn, Shanghai Jiao Tong University
Chuipin Kong, kongchuipin@163.com, Shanghai Jiao Tong University
Xionghui Zhou, xhzhou@sjtu.edu.cn, Shanghai Jiao Tong University

Keywords:
Part Library, Cross-platform, Service, Parametric Design, ACIS

DOI: 10.14733/cadconfP.2017.283-287

Introduction:
With the rapidly increasing needs in product customization and specialization, a limited series of general specifications of industrial merchandises can no longer well satisfy various customers' requirements. As an indispensable foundation, large number of parts are necessary to be applied in product assembly. To expediently and agilely access to the parts information, several part libraries or similar systems have been presented, some of which are developed by part suppliers to promote their goods. Meanwhile, some manufacturing enterprises also establish independent modules or subsystems in ERP for an effective management of the parts from different suppliers [1-5].

From the view of detailed implementation, the part library systems construct their applications on a certain CAD platform to effectually utilize the built-in APIs or existing functions. Thus, much work can be saved during the constructing period, while on the other hand the libraries inevitably depend heavily on the exact host CAD platform and cannot be transplanted onto another. Therefore, the part libraries are hard to be extended, transplanted or shared by other systems in the stage of upstream and downstream partners. As a result, the numerous part libraries are enclosed to form several isolated islands and doesn't hold the ability of communicating with each other. Also, information redundancy, mutual contradiction or repeated developments will frequently trouble the designers or enterprises. In addition, they rarely support the idea and mechanism of providing independent function alone, that is, without any support from CAD platforms.

Aiming at solving these problems, this paper investigates a service oriented cross-platform (or namely, CAD platform-independent) framework for 3D part library. The term "service" in the context means that its implementation is open to any individual or organization, and is to bridge the gaps amongst users, CAD platforms or other host systems, with which the users can expediently use, add, delete and edit the information for their own application. Moreover, the method to implant design knowledge into the parts is also investigated. Flexible and effective representation of diversiform knowledge is an important issue in the reuse of existing geometrical, structural, or logical information in the stage of model edit, which is also emphasis in our work.

Architecture & Methodology:
To realize the purpose, we have to construct the applications without the support from the existing CAD systems to avoid the compulsive pre-requisite of their installations and all the functions are implemented in our owned way or from a lightweight API packages.

The architecture of the proposed platform is illustrated in Fig.1, in which the kernel module is part library service (PLS). Through the adapters to establish the connections between PLS provider and various kinds of host systems, it can support several kinds of functions such as 3D part modeling and preview, type navigation, BOM, etc.

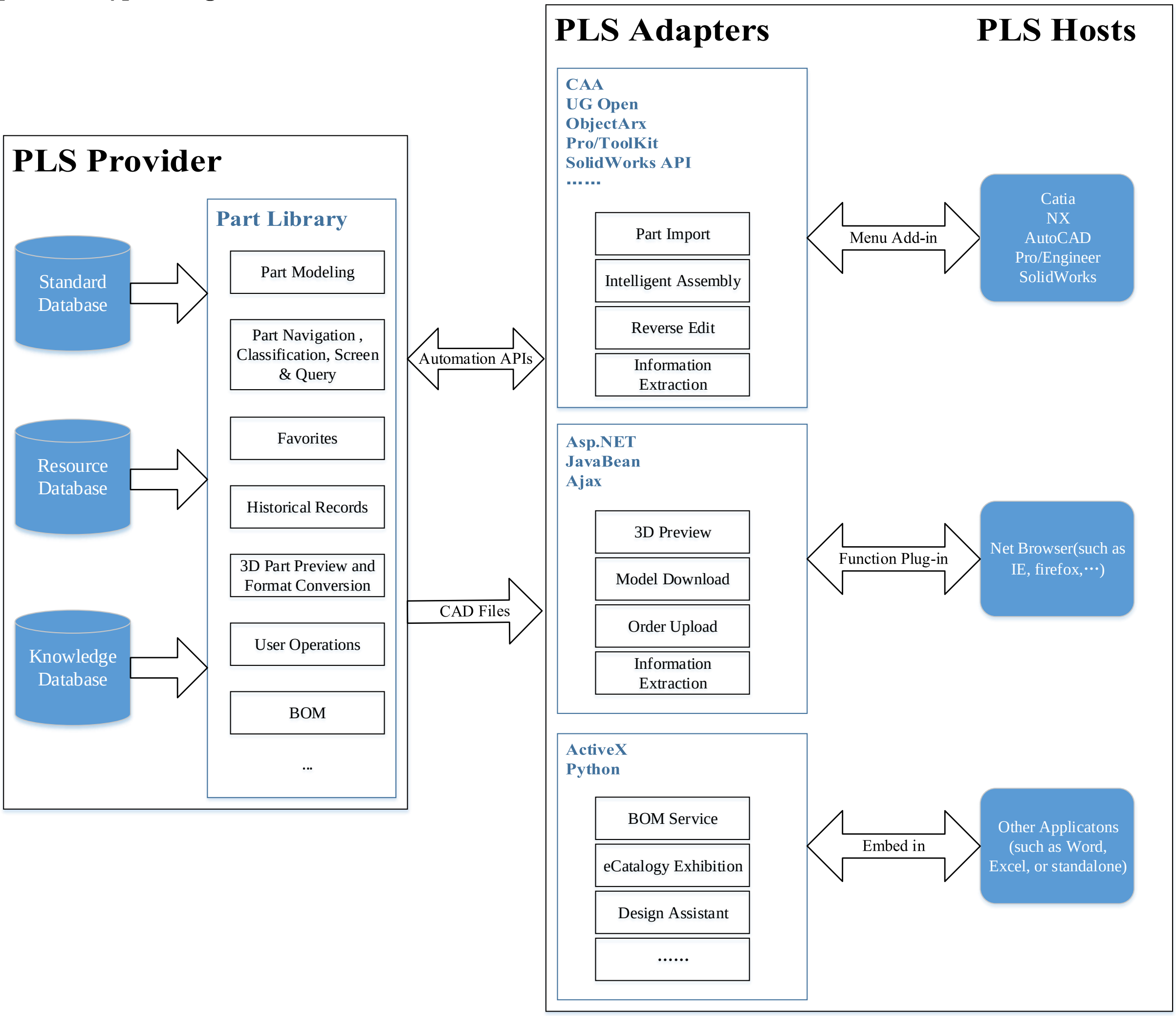

Fig. 1: The architecture of the part library service system.

The PLS provider can be divided into 3 subsystems: type/parameter selection, model generation/ drive and PLS APIs. The first subsystem is more known as part e-catalog, and the second one is for the uses of 3D part model generation in various formats according to the actual CAD design though it is implemented on a non-CAD environment. The exported functions of “service” is realized by PLS, a package of APIs to be invoked by any other applications on certain criterions through the adapters, or connectors, which connect PLS with host systems, especially with the general CAD platforms such as NX, Pro/E, AutoCAD, Catia, SolidWorks, etc. In addition, the main functions can also be wrapped and embedded in the website through Asp.net or Ajax.

To our knowledge, the PLS method is currently seldom supported in the field of part library since most congeneric systems are of introverted environment, which limits the applications to a relatively narrow usage occasions.

Part Information Management

The system provides various methods to locate the actual part. The most common way to access a part is by a three-level “category-type-parameter” navigator as shown in Fig.2.

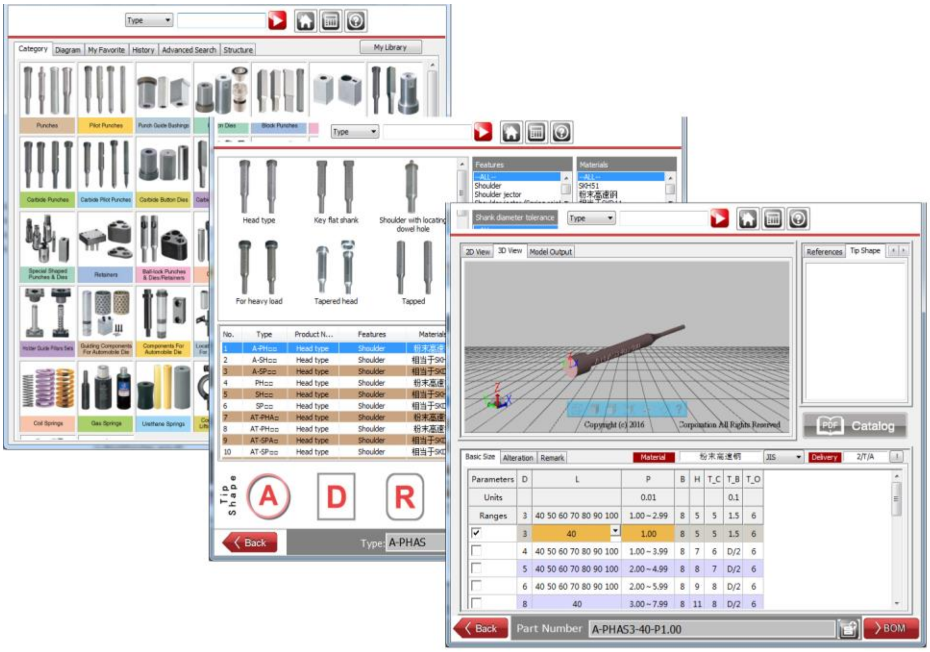

Fig. 2: The three-level navigator.

Besides the navigator, users can access the part through search by type, remark, keywords, etc. The system also records the part selection information, based on which remark, BOM, my favorites and history record can be established.

Remark: a serial of tooltips on the advantages and disadvantages, key points, which are attached to an individual part.

BOM: the part information about the products. BOM also can be created by the design information in CAD systems through traversing the assembly trees. The information is stored in the database and can be exported into Excel files according to the pre-defined template, which is easily customized.

My favorites: the shortcuts to access the most commonly used parts with all parameters reserved in the personal account for each designer.

History record: the shortcuts to access the previously visited parts with all parameters reserved in the personal account for each designer.

3D Part Modeling and Format Conversion

Without the support from built-in CAD systems, the real-time 3D preview and update of the parts is a great challenge. In our implementation, the 3D part models are created with ACIS, a famous geometry engine. To ensure the parameters list can construct a valid combination, a pre-check mechanism is embedded in the system. That is, although the geometric sizes of a body are in a complicated set of constraints and relations, the user will achieve a correct list however he inputs or selects parameters as the knowledge about dependency relations and rules have been inherent in the systems.

The parameters are input into a modeler, in which .sat file are created and InterOP and RealDWG are then adopted to convert it into various formats including .stp, .iges, dxf, dwg, hsf, etc. The preview of 3D model in the interface is based on Hoops with .hsf file.

Geometry and rule constraints based on diversiform knowledge

As the sizes in geometric shape of any part are interrelated and share many complex constraints, it is necessary to deeply investigate the effective way to represent and realize the relations. In this paper, we adopt various kinds of knowledge to handle the problems.

Firstly, the parameter linking tables are the uppermost method to store and handle the geometric relation. The parameters are divided into active and passive ones, and if an active one is change, the range, value, or even precision of its associated passive ones will also be corresponding changes. An expression evaluator is embedded to compute the actual value and precision of each parameter. Fig.3 illustrates a parameter table for just one part, in which all parameters are associated.

D	L	P-0.01	W-0.01	A-0.01	B-0.01	R-0.01	S-0.01	Kmax	Wmin	bb	B1	dd	F	K_check-0.01	T_O	T_C	T_B-1
10	16 20 22 25 30 35 40	2.82 ~ 6.00	2.01 ~ !P	1.01 ~ W-1	1 ~ !A	0.2 ~ 40	0.2 ~ !(P-W)/2-R,40	6.00	2.00	10	6	6.4	6	sqrt((P-2*R)*(P-2*R)+(A-2*R)*(A-2*R))+2*R ~ Kmax	1	1	D/2
13	16 20 22 25 30 35 40	2.82 ~ 8.00	2.01 ~ !P	1.01 ~ W-1	1 ~ !A	0.2 ~ 40	0.2 ~ !(P-W)/2-R,40	8.00	2.00	10	6	8.4	7.5	sqrt((P-2*R)*(P-2*R)+(A-2*R)*(A-2*R))+2*R ~ Kmax	1	1	D/2
16	16 20 22 25 30 35 40	3.31 ~ 10.00	Wmin ~ !P	1.01 ~ W-1	1 ~ !A	0.2 ~ 40	0.2 ~ !(P-W)/2-R,40	10.00	2.50	10	6	10.6	8	sqrt((P-2*R)*(P-2*R)+(A-2*R)*(A-2*R))+2*R ~ Kmax	1	1	D/2
20	16 20 22 25 30 35 40	3.82 ~ 12.00	Wmin ~ !P	1.01 ~ W-1	1 ~ !A	0.2 ~ 40	0.2 ~ !(P-W)/2-R,40	12.00	3.00	12	8	12.6	10	sqrt((P-2*R)*(P-2*R)+(A-2*R)*(A-2*R))+2*R ~ Kmax	1	1	D/2
25	16 20 22 25 30 35 40	4.82 ~ 16.00	Wmin ~ !P	1.01 ~ W-1	1 ~ !A	0.2 ~ 40	0.2 ~ !(P-W)/2-R,40	16.00	4.00	12	8	16.6	12.5	sqrt((P-2*R)*(P-2*R)+(A-2*R)*(A-2*R))+2*R ~ Kmax	1	1	D/2
32	16 20 22 25 30 35 40	5.82 ~ 20.00	Wmin ~ !P	1.01 ~ W-1	1 ~ !A	0.2 ~ 40	0.2 ~ !(P-W)/2-R,40	20.00	5.00	15	10	20.6	16	sqrt((P-2*R)*(P-2*R)+(A-2*R)*(A-2*R))+2*R ~ Kmax	1	1	D/2
38	16 20 22 25 30 35 40	6.82 ~ 26.00	Wmin ~ !P	1.01 ~ W-1	1 ~ !A	0.2 ~ 40	0.2 ~ !(P-W)/2-R,40	26.00	6.00	15	10	26.6	19	sqrt((P-2*R)*(P-2*R)+(A-2*R)*(A-2*R))+2*R ~ Kmax	1	1	D/2
45	20 22 25 30 35 40	6.82 ~ 35.00	Wmin ~ !P	1.01 ~ W-1	1 ~ !A	0.2 ~ 40	0.2 ~ !(P-W)/2-R,40	35.00	6.00	20	14	36.0	22.5	sqrt((P-2*R)*(P-2*R)+(A-2*R)*(A-2*R))+2*R ~ Kmax	1	1	D/2
50	20 22 25 30 35 40	7.82 ~ 40.00	Wmin ~ !P	1.01 ~ W-1	1 ~ !A	0.2 ~ 40	0.2 ~ !(P-W)/2-R,40	40.00	7.00	20	14	41.0	25	sqrt((P-2*R)*(P-2*R)+(A-2*R)*(A-2*R))+2*R ~ Kmax	1	1	D/2
56	20 22 25 30 35 40	8.82 ~ 45.00	Wmin ~ !P	1.01 ~ W-1	1 ~ !A	0.2 ~ 40	0.2 ~ !(P-W)/2-R,40	45.00	8.00	20	14	46.0	28	sqrt((P-2*R)*(P-2*R)+(A-2*R)*(A-2*R))+2*R ~ Kmax	1	1	D/2

Fig. 3: Parameter table for a part.

Second, the rules of parameter and alternation selection (as shown in Fig.5) are also stored in the knowledge database with an editing tool(KDT) for the users in the format of "IF-THEN" expresses. KDT supports an active monitoring for the change of parameters or alternation, with which if their inputs or selections invoke certain rules, the later will take effect and KDT will automatically take corresponding per-defined measures to insure the data correction.

Third, the backward edit and associated assembly will send the current design information back to PLS provider, which also assists PLS to automatically screen suitable parameter lists for 3D modeler. In this case, the PLS is more like a knowledge receiver and convertor from the real designs.

Service in CAD platforms:

Through different kinds of PLS adapters, the PLS provider can be deployed in the hosts. CAD systems are the most common users. Fig.4 is two hosts as AutoCAD and Catia.

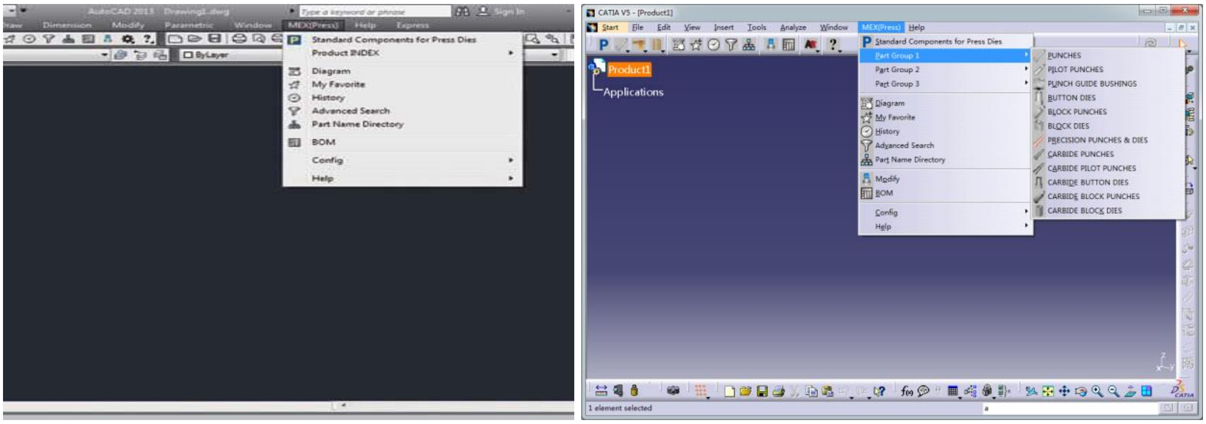

Fig. 4: PLS provider hosted in CAD system, (a) AutoCAD, (b) Catia.

The PLS adapter are developed with the API packages provided by the host systems and invoke the functions exported from PLS provider. Integrated in the CAD system, the created 3D part models can be directly inserted into the design with flexible assembly methods, as illustrated in Fig.5

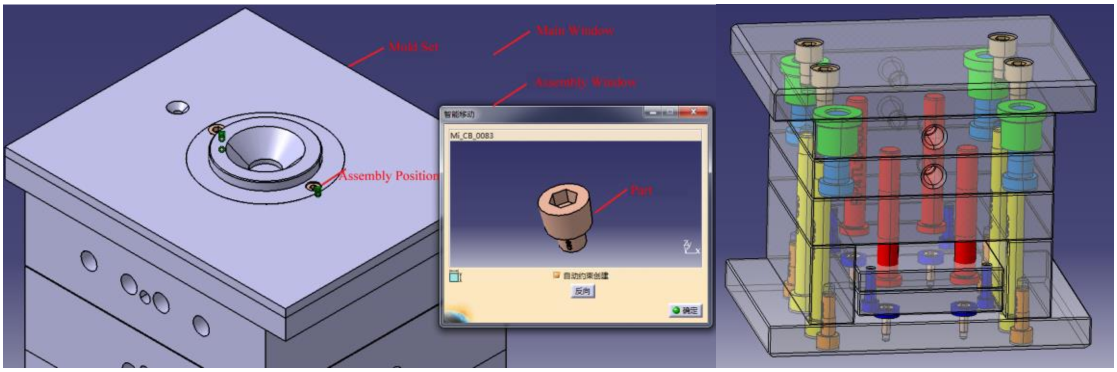

Fig. 5: Mold Set Design with PLS system in Catia.

Conclusion:

The system has been successfully applied in many manufacturing enterprises and brings great benefits to them. Compared with the traditional standard part library system, this system holds the following characteristics:

1) Cross-platform: The system doesn't depend on a specific CAD platform. The database module and the CAD operating module are separated. An independent CAD modeling kernel is equipped in the system. A rapid expansion and application in different CAD platform is realized to avoid developing several specific systems for different CAD platforms.

2) Network sharing: A unified network-based database system is established to facilitate a unified management and update of enterprise data and knowledge, which is conducive to teamwork.

3) Convenient operations: The system is designed for die and mold design process according to the user's operating habits. Optimization functions are provided for the design process, which greatly enhances the user's design efficiency.

The system also holds some shortcomings needed by improved. Firstly, the 3D models are created with ACIS and converted into .stp or .iges, not the native format, so the imported parts only contain information with geometry and attributes, while its features and structure is missing, which brings the Inconvenience in modification. Direct driving of native model in CAD systems is an interest in our later research. Another, the types in part library cover only mold and die, other kind of parts should also be taken into consideration. As for the further work, we plan to extend the part library from die and mold to more kinds of parts such as factory automation in manufacturing industry.

References:

[1] Fan, Q.-M.; Liu, G.; Wang, W.; Development of Die Sets standard parts library based on Pro/E. Procedia Engineering, 15, 2011, 3802-3807. http://dx.doi.org/10.1016/j.proeng.2011.08.712

[2] Jong, W.-R.; Wu, C.-H.; Li, M,-Y.: Web-based navigating system for conceptual mould design with knowledge management, International Journal of Production Research, 49(2), 2011, 553-567. http://dx.doi.org/10.1080/00207540903439830

[3] Lee, J.-H.; Kim, S.-H.; Lee, K.: Integration of evolutional BOMs for design of ship outfitting equipment, Computer-Aided Design, 44(3), 2012, 253-273. http://dx.doi.org/10.1016/j.cad.2011.07.009

[4] Ma, Y.-S.; Tor, S.-B.; Britton, G.-A.: The development of a standard component library for plastic injection mould design using an object-oriented approach, The International Journal of Advanced Manufacturing Technology, 22(9-10), 2003, 611-618. http://dx.doi.org/10.1007/s00170-003-1555-8

[5] Mok, H.-S.; Kim, C.-H.; Kim, C.-B.: Automation of mold designs with the reuse of standard parts, Expert Systems with Applications, 38(10), 2011, 12537-12547. http://dx.doi.org/10.1016/j.eswa.2011.04.040

Proceedings of CAD'17, Okayama, Japan, August 10-12, 2017, 283-287