Computer-Aided Design and Applications

http://www.cadanda.com

Targeted WebCAD

Gerardo Salas Bolaños[1], Stephen Mann[2] and Sanjeev Bedi[3]

[1]University of Waterloo, gsalas@engmail.uwaterloo.ca
[2]University of Waterloo, smann@cgl.uwaterloo.ca
[3] University of Waterloo, sbedi@mecheng1.uwaterloo.ca

ABSTRACT

CAD software is increasingly complex software capable of a large variety of tasks. Unfortunately, this makes it a poor match for specific small or targeted manufacturing industries, since to use a parametric solid modeler, a company has to employ one or more designers to learn and use software of which only a small part is relevant to the industry. Such costs make it prohibitive for small companies to effectively use high end CAD software in their design systems. As an alternative, we envision web based software targeted at specific industries. Rather than expose a complete parametric solid modeler, the web interface is targeted to the specific manufacturing industry, with design tasks specialized to the object of manufacture. We have implemented and deployed such a system for table leg manufacturing. In this talk, we give an overview of this implementation and discuss the issues that arise in developing this type of CAD model.

Keywords: solid modeling, web based, manufacturing.
DOI: 10.3722/cadaps.2009.639-644

1. INTRODUCTION

1.1 Background Information

The capability of performing many tasks enables CAD software to design complex parts. While this is an advantage for large manufacturing industries, it makes it a poor match for small or targeted manufacturing industries. These industries cannot justify the cost of employing one or more designers to use software for which only a small part is relevant to their industry. In addition, the designer requires extensive training to use a parametric solid modeler given its complexity. These costs make it unaffordable for small companies to effectively use high end CAD software in their design system. An alternative for small targeted manufacturing industries is web-based software that will allow the design of a specific product. This software is created using the programmable geometric algorithm capability provided by solid modelers.

The World Wide Web (WWW) has become the most popular information system on the Internet. As it continues to expand it is shifting from its original concept of tracking scientific data in a project to a wide range of services such as news, virtual shopping, and social networking utilities [8]. The CAD industry has not taken full advantage of the benefits of the Internet such as distributed computing. In distributed computing the user's computer sends data to a remote machine, which returns information that is displayed on the local machine. Remote processing enables a user with a low end system to process data remotely on a high end system and visualize the results locally on the on a low-end system. Distributed computing can be used to run web-based CAD software on any computer; the user

enters data into the computer which is sent to a remote server connected to the Internet. This server does all the calculations and runs geometric algorithms to generate a solid model based on the data the user entered. The server returns data to the user. The computer uses this data, does some client-side processing, and generates two and three dimensional views of the object being generated.
The data that is entered by the user consists of simple parameters related to the product being designed. This data alone is not enough for the server based parametric solid modeler to create a solid model. Geometric algorithms have to be used to convert the parameters into commands the solid modeler understands. Most solid modelers have an Application Programming Interface (API) that enables users to create geometric algorithms to automate tasks. The API can be used to create a series of product specific algorithms that will automatically create a product based on simple parameters entered by the user.
The complexity of a solid modeler is avoided by eliminating unnecessary tasks. The complete parametric solid modeler is not exposed; instead a web interface targeted at a specific manufacturing industry with design tasks specialized to the object of manufacture. By making the software web-based the cost of purchasing a full parametric solid modeler is eliminated for the user. For the developer there is also a benefit given that the cost of developing this framework is divided among all the users. Furthermore, the geometric algorithms make the program simple to use and makes the design process simpler.
In this paper the implementation of such a system will be discussed. This system has been targeted at the table leg manufacturing industry. The issues that arose in the creation of such a system will be described.

1.2 Related Research

In recent years the research in the field of web-based design has taken importance with the increasing use of the Internet. Regli [5] describes some of the different trends influencing CAD over the Internet. Different types of web-based CAD systems have been developed. Such is the case of Cybercut [2], an Internet-based CAD/CAM system that allows users to remove material from a rectilinear block to create a solid model and then automatically carries out the process planning required. Zeng et al. [9] developed a web-based CAD system that interfaces with the user using Java applets and allows the creation of basic curves and surfaces. Roy and Kodkani [6] created a framework that allows collaborative CAD over the World Wide Web. Nousch and Jung [4] created a program called BEAVER for a specialized task: the creation of closets. They have found that the web-capabilities and free cost of usage have been welcome in their industry.

2. DEVELOPEMENT OF TARGETED WEBCAD

2.1 Overview

WebCAD was designed using geometric algorithms that automate the design of table legs. It is a web based application that builds a geometric model of a table leg based on simple parameters provided by the user. The model created is then communicated to the client for final approval and inspection.
WebCAD can be accessed using any web browser. The web based front end consists of a series of forms in which the user enters simple design parameters. Each form represents the generalization of a specific feature. Once a form is completed by the user, the parameters are sent to the WebCAD server over the Internet. The design parameters are processed by the geometric algorithms, housed in the server, to generate a geometric model. The server then generates two and three-dimensional views of the geometric model and sends them back to the user. At this point, the user can make modifications to the model if required. The general process described is portrayed in the diagram shown in **Error! Reference source not found.**.
The advantages of WebCAD include.

1. **No installation or licensing fees** – WebCAD does not require installation of any solid modeler. Given that all the geometric algorithms and the solid modeling engine are housed in the WebCAD server, the only thing the user needs is an Internet connection and a web browser. This also means that expensive workstations are not required. By taking advantage of distributed computing all the advanced calculations are done on the server reducing the processing done on the client side.

2. **Ease of u** essible to workers in a particular

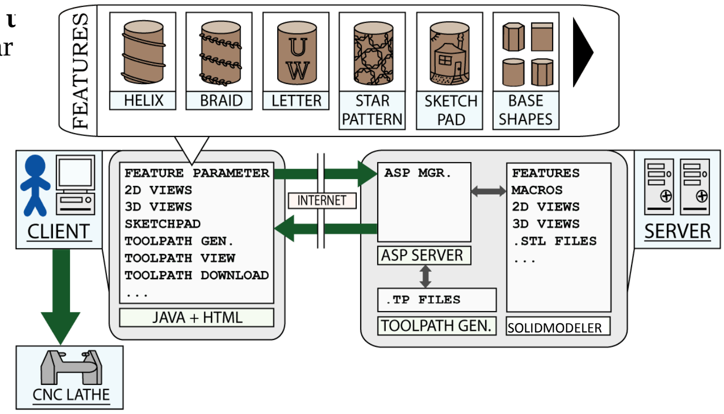

Fig. 1: Overview of WebCAD.

3. **Targeted software** – the WebCAD interface is targeted at the table manufacturing industry. This means that the features available are specific to this industry. Unnecessary features that are available in a commercial package are hidden from the user to reduce the complexity of the software.

The software was initially programmed using the VBA environment provided by SolidWorks, and then was ported to Visual Basic .NET (VB .NET) in the later stages. Programming using the SolidWorks VBA environment allowed for efficient geometric algorithm development and debugging of parameterized features for leg designing. Porting to Visual Basic .NET ensures solidity and convenient connectivity to ASP .NET for the development of the web based front-end. Visual Basic was chosen as the programming language because SolidWorks macros use Visual Basic scripting.

2.2 System Configuration

The architecture of the system is shown in Fig. 2. It is made of six modules, namely: the web browser on the client side and the ASP Manager; geometric algorithms; solid modeler; graphics generator; and database on the server side. The web browser interface is responsible for the interaction with the user. It is in charge of getting the input from the user and sending to the server. It also receives the data from the server and display the geometric model. This is the only module that is visible to the user.

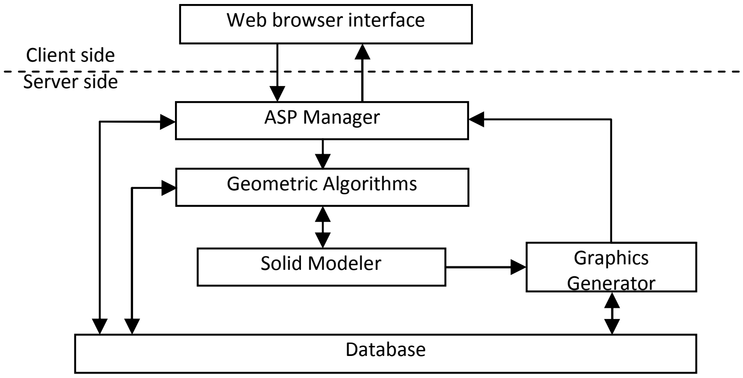

Fig. 2: WebCAD system architecture.

The ASP manager is responsible for receiving the data from the user interface and communicating with the other modules in the system. This module communicates with the database and the geometric algorithms. The data that is received in the ASP manager is updated in the database. Once the data has been updated, the ASP manager sends the data to the geometric algorithms to start generation of the solid model.

Once the geometric algorithm module receives the data it starts generating the solid model along with the solid modeler. These two modules have to interact to generate the solid model. The calculations are carried out in the geometric algorithm module. The geometric algorithm module then sends the result to the solid modeling module and must wait for this module to send a response that it is done generating the current feature so that more data and additional steps can be carried out.
When the geometric and solid modeling algorithms are done generating the solid model, they send it to the graphics generator. This module is responsible for generating the two and three-dimensional views of the solid model. The two dimensional views are pictures with JPEG-format file. The three dimensional view is created using an STL-format file and OpenGL. The graphics module communicates with the ASP manager that sends the data back to the client side for viewing.
The database module is made up by sub-databases. Each sub-database is assigned to a user. It records all the information and parts generated by a specific user. Given that the geometric algorithm module cannot carry out calculations for unlimited number of users, the database module is used to store the data for a user while the geometric algorithm module generates the data for another user.
The data flow throughout the framework is shown below.

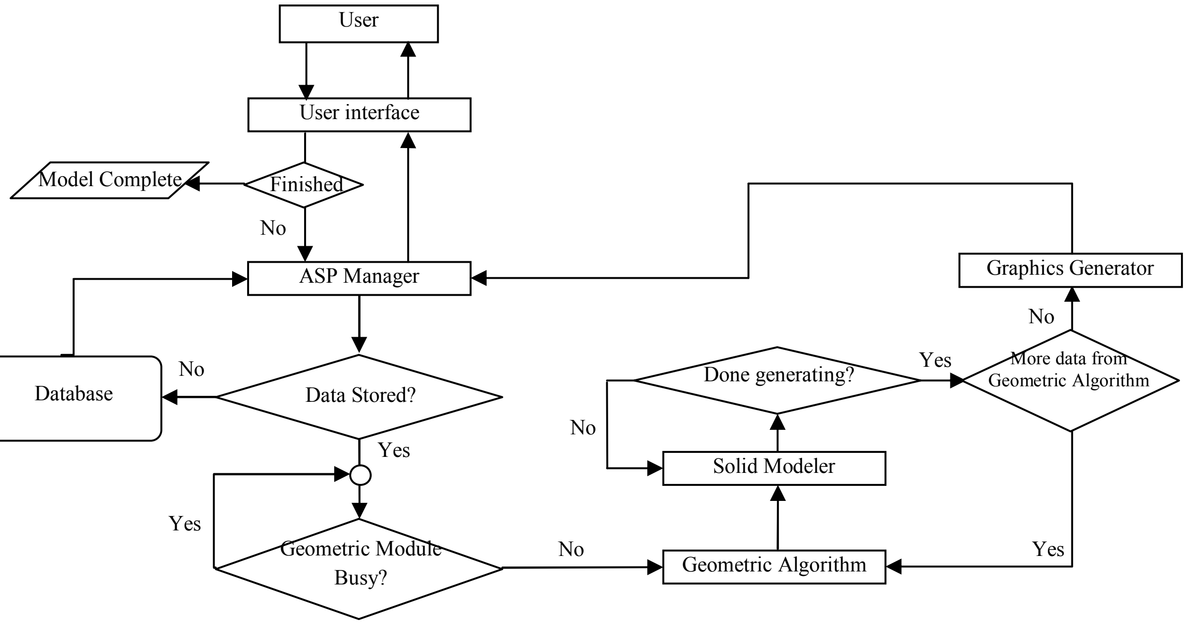

Fig. 3: WebCAD data flow.

2.3 Identification of Design Parameters
One of the most important steps in the development of this software was determining the design parameters of the product being designed: table legs. It is important that this be determined to decide what features of the solid modeler are going to be exposed in the web interface. It is important to allow the most design flexibility without making the software too complex.
In the table leg industry there are three main types of products: dining table legs, end table legs, and coffee table legs. The main difference between the three types of table legs is their length; dining table are legs 29 inches, end table legs 21 inches, and coffee table legs 18 inches.
Despite having different lengths, the elements that make up the legs are the same. As can be seen in Fig. 4, five sections make up a leg. The first section (a) is the connector; this part of the leg is attached to the table surface. This section is usually a square prism, although for pentagon or hexagonal tables the section may have five or six sides respectively. In some designs, small carvings may be added to this section.

The second section (b) in a table leg is a transition section. This section serves as a connection between the connector section and the main body of the leg. The shape of this section varies with design. Usually this section is made up by rings on a cylindrical body.
The main body of the leg (c) is the longest section in a leg. The designs on this portion of the leg vary significantly. Designs can be as simple as a cylindrical main body or as intricate as an array of helices on solid of revolution.
The fourth section (d) is again a transition section. In this case it is a transition between the main body of the leg and the foot of the leg.
The last section of the leg (e) is the foot. This is the section of the leg touching the floor. This section is usually made up by a solid revolution of a profile.

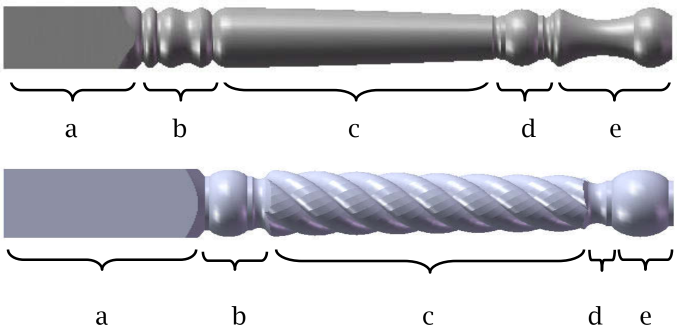

Fig. 5: Table leg sections: a. connector, b. transition, c. main body, d. transition, and e. foot.

The length and shape of each section is not fixed. Furthermore, some leg designs may be missing transition sections. Due to this it was decided that the best method for designing was a totem pole style. First the user must decide how many sections the leg is going to have. Then each of these sections must be designed based on the length, diameter, and cross section shape. These sections are called the *base sections*. The user can then add features onto each one of the base sections. The features that are available are: helices, rings, braids, lines, lettering, and star patterns.

3. IMPLEMENTATION

As mentioned in the previous section of this paper the chosen method for designing the legs is a totem pole approach. For the implementation of this, a main class, *Leg Base*, which holds all the information about a particular leg was created. The class object declared by *Leg Base* is *MainLeg*. Various sub classes called *LegSection* make up the *MainLeg* object. The *LegSection* class stores all the add-on features contained in a leg section along with leg section base profile information. Users are allowed to access one leg section at a time for adding features. This organization increases the ease with which the user can add, modify, or delete features in a leg section.
The add-on features for the leg sections each have their own class. These classes have access to the information of the specific leg section and create the geometry based on this information. While the geometry created by the add-on feature classes is dependant on the geometric characteristics of the leg section, it is independent of other features on the leg section.
The hierarchy described in this section is shown in the figure below.

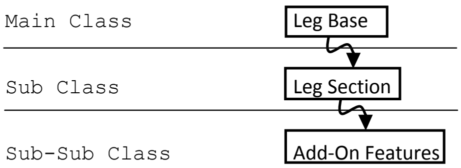

Fig. 5: Hierarchy of classes for WebCAD.

The architecture of the system is such that it can be used for any product not just table legs. The framework does not require modifications, only the geometric algorithms for the base and the add-on features must be modified given that these are specific for each type of product.
The geometric algorithms for the add-on features have been programmed so that they can be added to any of the available leg section profiles. The algorithm used is the same for any type of leg section, but by accessing the leg section's information an appropriate feature is created. For example, the ring feature is a donut shaped when it is built on a cylindrical leg section when. If it is added to a square section then it will not be donut shaped, the ring will follow the square profile. The path used to sweep a circular profile is a circle in the case of a cylindrical section and a square in the case of a squared profile cross section. The geometric algorithms must be generic and robust enough to allow for different base sections.

4. CONCLUSION

We have described WebCAD, a web-based system for the design of table legs. The creation of this system and its acceptance in the table leg industry prove that targeted CAD software is a viable option for specific industries. By only exposing the required features the complexity of a complete CAD package is eliminated and the result is user-friendly software.
This system has been targeted at the table leg industry but it can easily be modified to allow the design of almost any product. The same framework that has already been created can be used; the only modifications required would be in the type of base shapes and features that can be added to this base shape. This framework can be used as a template to create any targeted CAD system, it has already been successfully used in a web-based system for the sign making industry.
The web-based approach has also received positive feedback. The savings in cost in a per use payment scheme of the web-based system in comparison with software purchasing and licensing associated with complete CAD packages have made this system attractive to many industries.

5. REFERENCES

[1] Adamczyk, Z.; Kociolek, K.: CAD/CAM technological environment creation as an interactive application on the Web, Journal of Material Process Technology, 109, 2001, 222-228.

[2] Ahn, S. H.; Sundararajan, V.; Smith, C.; Kannan, B.; D'Souza, R.; Sun, G.; Mohole, A.; Wright, P. K.; Kim, J.-H.; McMain, S.; Smith, J.; Sequin, C. H.: CyberCut: An Internet-based CAD/CAM System, Transaction of the ASME, 1, 2001, 52- 59.

[3] Huang, G. Q.; Mak, K. L.: Design for manufacture and assembly on the Internet, Computers in Industry, 38, 1999, 17-30.

[4] Noush, M.; Jung, B.: CAD on the World Wide Web: Virtual Assembly of Furniture with BEAVER, ACM, 1999.

[5] Regli, W. C.: Internet-enabled computer aided design, IEEE Internet Computing, 1, 1997, 39-50.

[6] Roy, U.; Kodkani, S. S.: Product modeling within the framework of the World Wide Web, IIE Transactions, 31, 1999, 667-677.

[7] Silva, M. J.; Katz, R. H.: The Case for Design Using the World Wide Web, 32nd ACM/IEEE Design Automation Conference, 1995.

[8] Smith, C. S.; Wright, P. K.: Cybercut: A World Wide Web Based Design-to-Fabrication Tool, Journal of Manufacturing Systems, 15 (6), 1996, 432-442.

[9] Zeng, J.; Chen, W.; Ding, Q.: A Web-based CAD system, Journal of Materials Processing Technology, 139, 2003, 229-232.