UNDERGRADUATE REPORT

The Intelligent Process Planner and Scheduler

by Carl P. Thompson
Advisor: Jeffrey W. Herrmann, Edward Lin, Mark Fleischer, Vidit Mathur

U.G. 2000-1

ISR

INSTITUTE FOR SYSTEMS RESEARCH

The Intelligent Process Planner & Scheduler

By Carl Thompson
Georgia Institute of Technology
Atlanta, Georgia

Advisors
Dr. Jeffrey W. Herrmann
Dr. Edward Lin
Dr. Mark Fleischer
Vidit Mathur

Research Experience for Undergraduates
Institute for Systems Research
University of Maryland
College Park, Maryland

August 4, 2000

Table of Contents:

I.   The REU Program

Research is an opportunity to broaden the intellectual, conceptual, and technological horizons of everyone involved. Research experiences are typically enjoyed only at the graduate level of education. Graduate students have a wealth of available research projects to work on because of their experience level in a particular subject. In some graduate programs, research experience is mandatory because of the wonderful benefits students can take from research experiences. Through grants from the National Science Foundation, numerous universities have been able to create programs called Research Experiences for Undergraduates. These programs provide an outlet for undergraduate students to share in such an important educational growth experience as research.

II.   Introduction

Information relevant to a particular system is similar to the energy of a system in that it is continually being released as a system progresses. In the past science has focused on harnessing the energy released by a system and using it for a particular cause, like to make electricity. This is the fundamental concept of a power plant. Today in the information age, scientists are now interested in a research thrust that involves harnessing and effectively utilizing the information released by a system. Research trends such as information and energy that deal not with the fundamentals of a system but a system's results stem from scientists' ability to capture those results. The capture of energy was made possible by various energy converters like the windmill and the nuclear reactor. The capture of information has been made possible by the computer.

The research thrust today is not how do we store information, the question is, then what? What can we do with information such as history or runtime data, to improve the way we operate a system?

III.   Problem

Low volume production typically associated with military space systems manufacturing is inherently expensive, time consuming and risk laden. This state of manufacturing systems is caused by an inability to adequately predict, monitor, and control the product development and manufacturing process. This results from the lack of consistent flow capture, and management of the information available within an enterprise.

What the ASSIST project is about is creating an intelligent information management system (entitled ASSIST) that incorporates proven technologies for seamless integration and dissemination of information amongst the entire enterprise. The Institute for Systems Research (ISR) at the University of Maryland (UMD) is responsible for the information management tool that handles the planning and scheduling component of the manufacturing system. The tool is called the Intelligent Process Planner & Scheduler (IPPS).

There have been many software applications such as ASSIST that focus on project management. Microsoft Project is one example. Applications such as these focus mainly on managing information and

less on dissemination of the information. The key factor of ASSIST and its component applications is that they are Web-based. Web-based applications like ASSIST are the applications that will drive e-commerce into the future. The internet itself has proved to be a wonderful *access* tool. However the internet itself provides no way to organize its endless wealth of information in a understandable and easy to find manner. Web-based applications combine information management tools with the internet. This combination will not only be able to harness and manage information effectively, it will also be able to disseminate large amounts of information to proper resources, either automated or human, in an organized and understandable fashion. This procedure is called the optimization of information flow.

IV.   Project Background

The ultimate customer of the ASSIST is the United States Air Force. The Air Force has specified that the enterprise that ASSIST is to be modeled after is a Satellite Manufacturing Enterprise. Lockheed Martin is the general contractor and the University of Maryland Institute for Systems Research has been subcontracted work for IPPS.

V.   Web-based Applications: The Development Tools

The majority of web-based applications are written in Java. The reason for this is that the Java language is platform independent. Therefore an application written in Java can be hosted on any web server regardless of its platform. Another advantage of using Java is Java's capability of working with the internet.

As a student, taking intro to computing classes, I was accustomed to writing all code in notepad. What work experience has taught me is that most code is written using development environments. Development environments save developers a lot of time, and provide logic to the code writing process for beginning developers. IPPS was developed using IBM's web-application development software. This software package consists of two applications, Visual Age and Web Sphere. IBM Visual Age is an environment for writing Java code such as servlets. IBM Web Sphere is a very user-friendly HTML/JSP development environment. Compared to writing code in notepad, the IBM software is a superior development tool. I am unaware of how IBM compares with other development tools because IBM is the only one I have used to date.

VI.   Web-based Application: The Architecture

So how does a developer take an effective information management tool and put it on the web? For one, the software sits on a web server instead of in a computer's C drive. Web applications have 2 very fundamental differences from traditional applications. The software is installed on a web server, and the user interface exists on a web browser. The details of web-application architecture can be broken up into

three categories, the server side, the client side, and in between. Below is a graphical representation of these three elements.

The Basic Web Interaction Model
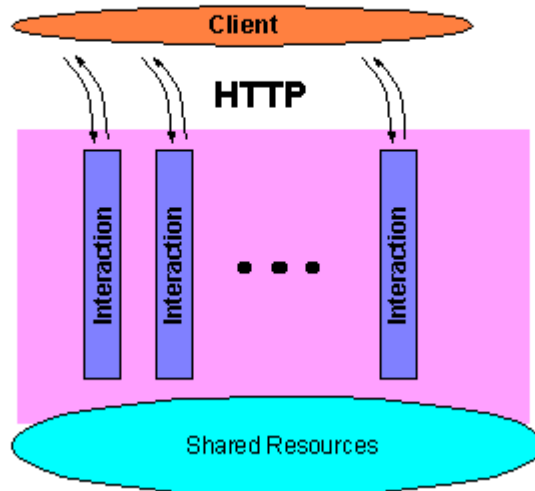
Since a server is just browser accessible memory space, describing the server side of a web-application really requires describing what sits on the server. All web-application servers contain a directory structure which holds Servlets, Java Server Pages (JSPs), HTML files, Java beans. Most web applications consist of much more, however these four items are fundamental to most application's functionality.

Servlets Servlets are Java programs that perform the work of different pieces of an application. The figure below shows the initial interface of IPPS. Each of the links in the left menu invokes each servlet of the web application.
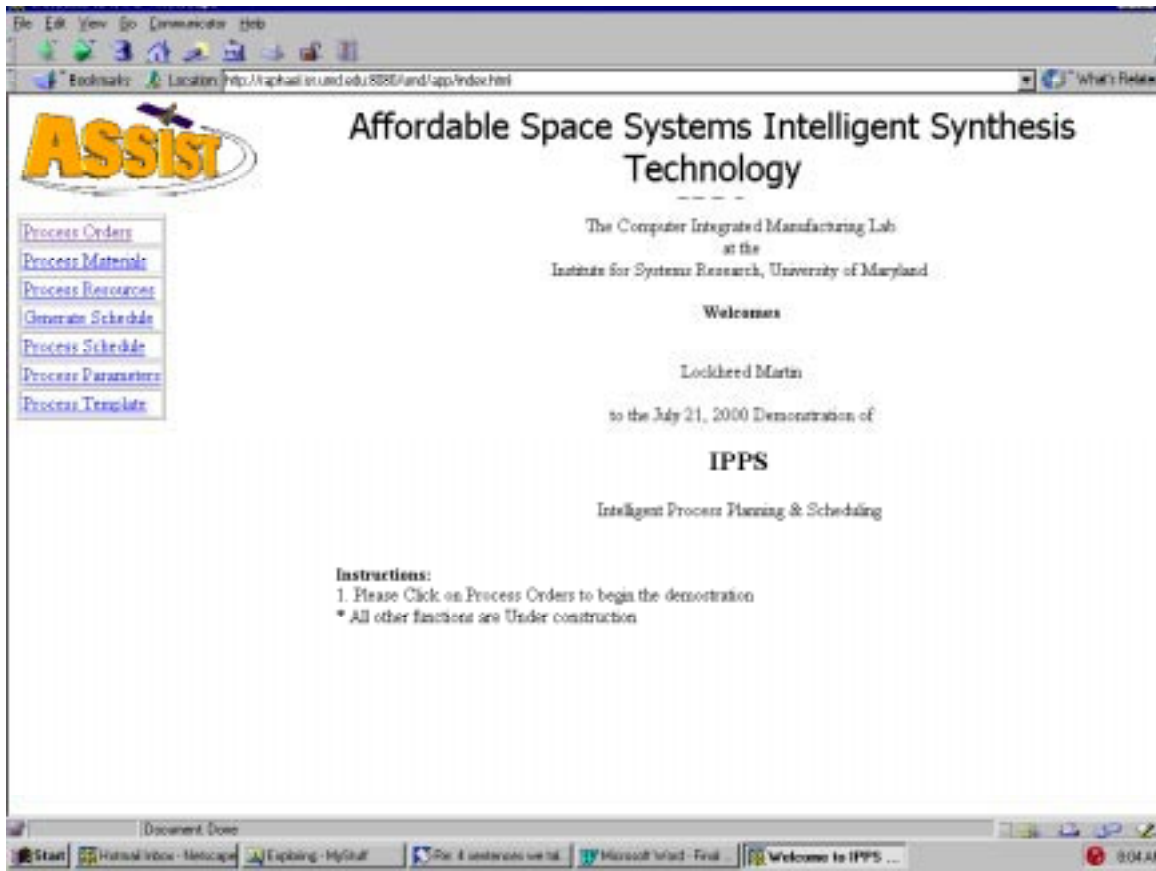
Just like non-web based applications, the code does not run by itself. An interface is used to clearly depict for a user what the application is capable of, and the user uses the interface to tell the code what to perform. Upon selecting the action the user wants to perform, the servlet is contacted via an HTTP request. The request contains the server name, followed by a colon; the directory structure component that leads to the proper servlet, followed by a question mark; and a parameter to inform the selected servlet of its immediate responsibilities. A sample HTTP request would look like this:

http://kalamazoo.isr.umd.edu:8080/ipps/ProcessOrdersServlet?action=DisplayOrders

The servlet code then performs the requested action and returns a resulting interface via a Java Server Page.

JSPs Creating JSPs was my direct area of involvement in the project. As stated above, users interface with web-based applications via a web browser. In early web-applications the most cumbersome part was creating all the static HTML pages to make an interface. What developers realized is that many of the HTML pages shared basic elements, with slight variations in content. For instance, the IPPS interface consists of a page that displays a list of all the satellite orders that await manufacturing. Using static HTML to create an interface would require making a page for a list of one order, a list of two orders, a list of three orders and so on. This task can be made much easier by using Java server pages. A JSP is an HTML page extended by a number of mechanisms to allow dynamic content to be added to the page as it is sent to the client (IBM 6). A JSP uses a Java bean. The creative use of Java beans in HTML pages allows room for

variables to be used in HTML pages. An example of where this functionality is useful is on a page that displays the properties of any item in a list. Here is a sample of what this would look like:

First Name	Last Name	Age	Birthday
Carl	Thompson	19	Dec 8 1980

A Java Server Page lets a developer create a static table with the proper headings, and then add variables for the actual values in the second row.

The IBM Web Sphere software was a vital resource in this stage of the development. Most developers are aware of the tedious code writing that is necessary to create an HTML page. Of course the code is complicated by the use of Java beans in HTML pages. The IBM Web Sphere development environment takes much of that work off of the web developer's shoulders. Making a table in web-sphere is just as simple as making a table in Microsoft Word. Taking adding dynamic elements in web-sphere takes some practice, as does learning any new software.

VII.   IPPS & the Extended Application

As mentioned earlier, most web applications consist of more than servlets and JSPs. A web application actually has the potential to leverage any application component on or off the web in order to perform its specific functionality. These additional components are defining characteristics that make a particular web-application unique. Some examples are databases that interact with the servelts and JSPs so data can be stored and accessed using the web, and decision support systems that leverage optimization tools so an enterprise can automate complex decisions and make the decision known throughout the enterprise.

The most distinguishing aspect of IPPS is its optimization engine which has the ability both to create a schedule, and to automate decision tradeoffs regarding planning and scheduling. Decision tradeoffs refer to mandatory, spur of the moment adjustments to the schedule caused by a personnel shortage or an incomplete material shipment. Factory managers currently have no way to evaluate these tradeoffs other than evaluating them in their head. The IPPS optimization engine uses mathematical operations research methods to analyze the results of these tradeoffs and determine the proper tradeoff to assure an optimal schedule emerges.

A manufacturing engineer operates this optimization by interfacing with a web browser. Under the Process Schedule functionality is a JSPs where the manufacturing engineer fills in information relevant to the scheduling process. The engineer can either select to create a new schedule or select an existing schedule that he chooses to modify. These parameters are passed to the server via an HTTP request. The Process Schedule servlet takes care of invoking the optimization engine, running the optimization, collecting the output, selecting a JSP to display the output and returning the JSP. This resulting JSP

displays the optimal schedule to the manufacturing engineer. The next step for the engineer is to press a button and export the schedule to the shop floor.

The ability to export the schedule to the shop floor is representative of a web based application's ability to disseminate information. Once the tradeoff evaluation has been made, resulting in a change, the change is not always made clear to all concerned parties in the manufacturing process. The benefits of IPPS and ultimately ASSIST are twofold. First is the tradeoff optimization and second is instant dissemination of real time information to all involved in the manufacturing system. These two ideas are the foundation for large amounts of research in e-commerce and information technology.

VIII.   Conclusion

I believe it is important for a student in the information age to be familiar with the concepts behind web technology. At times in our future we will be judged by how effectively we can retrieve information and process it into a solution. Exposure to the web is inevitable, yet with all the new technology that is constantly changing this newly discovered tool, it is often difficult to predict what the best way to locate relevant information may be. A quote by Benjamin Franklin (the first one to harness electricity) "the best way to predict the future is to create it." That quote may be the best argument for research experiences for undergraduates and for anyone interested in excelling in the information age.