Expert Systems for Online Advice

Knowledge at Your Fingertips

By Dustin Huntington

With the growth of e-commerce, the Web has become the primary conduit for businesses and customers to interact. At first the Web was a means of providing information — web pages substituting for printed brochures and email replacing phone calls. Downloads became the norm for software sales and updates. The ability to automatically convert virtually any standard form of document into HTML simplified Web page generation but resulted in so much data that finding specific information became significantly more difficult. Now, with the Web having grown so large, and the search engines being so well indexed, information overload is just a few keystrokes away. For those who want details and have the time to sift through pages of links, this is great, but for finding answers to specific questions this approach is not very efficient. One solution, the use of expert systems to deliver online advice, has been available for several years — with many noteworthy successes. When considering the implementation of an online expert system, however, certain issues need to be addressed to assure project success.

Providing Answers

Adding online calculators to Web sites was an initial attempt at providing assistance in the financial services area, where even the fairly simple calculation of compound interest, mortgage rates, etc., required special programs. Companies such as Numerical Objects (*www.numobjects.com*), through the use of Java, have simplified the addition of complex calculators to Web sites. If a problem or question only requires calculations, this is an efficient way to go.

Out of competitive necessity, companies are taking their Web sites to the next level, beyond information content, to providing online advice to help sell and support their products. Although HTML was created for the distribution of information, it was not designed for interactive advisory systems. However, companies are coming up with a variety of methods to implement the delivery of advice within the constraints of the Web.

Over the past few years the goal has been to provide more than just information but actual advice over the Web. The success of sites like Ask Jeeves (*www.askjeeves.com*) demonstrate the large demand for this type of user interaction. People want answers, not a list of sites. This is especially true of the growing percentage of computer neophyte's. However, it is just as true of the very Web savvy users who get the answers they want from one company and will fault others for not providing the same level of technology.

Logical Processing

The advice that companies want to deliver requires a higher level of logic than mere calculations. Often business problems require complex logical processing and probabilistic consideration of competing priorities.

Ideally, the Web site emulates the interaction with the best human sales or support team. This is where an expert system can be utilized, but as with any new capability, it must be implemented in a planned way.

Appropriate Problem

Having directly and indirectly participated in hundreds of expert system projects over the past 17 years, I have seen many become successful as well as those that are not so successful. There are many reasons why certain online advisory projects have failed; the most common are competing goals of the different groups involved in the implementation, and a lack of understanding of current technology limitations. Key factors to consider include project goals, complexity, interface and control.

Selecting an appropriate problem may seem obvious, but it continues to cause project failures. Expert systems are very good at handling well-understood, reasonably bound problems but since they don't think, they are not good at problems requiring intuition, personal taste or creativity. As an initial test, determine if the solution to the problem can be taught to a bright person in under a month — if it can't, then an expert system solution may not be practical. There are exceptions to this rule, but it is a good benchmark. Also, the problem should involve logical processing, usually of non-numeric data. If direct mathematical calculation provides the solution, it is better to use other technologies such as an online calculator. A professional knowledge engineer can help to scope a system and determine the best technology.

Appropriate Technology

Many forms and levels of advice can be added to a Web site Fuzzy databases, case-based reasoning and cognitive filtering tools are applicable when the advice can be based on statistical patterns from similar situations. These can be effective when the specific rules are not known, there is a large body of historical data, a best guess is adequate, and it is allowed to be wrong in some cases.

Expert system engines and development tools are designed for codifying specific complex knowledge as logical steps, and interacting with the visitor to determine what the logic implies about their particular situation. The expert system will give specific answers based on the rules. If the rules are complete and correct, the system will always give correct answers, but building a complex system that considers all situations is not necessarily easy.

A common misconception is that an expert system makes a site smart or intuitive. Expert systems are simply an efficient way to do logical processing — there is nothing they do that can not be done by conventional programming. However, an expert system development environment makes it much faster and easier to write, maintain and deliver knowledge advisory systems. And the inference engine is often the most practical way to handle the complex interrelation of logical elements that exist in real world systems.

As with any software development activity, it is far easier to purchase an established and proven engine than to develop one from scratch. Full-featured expert system development tools provide methods to build, test and validate systems through an interface that greatly speeds development and integration into the Web site. But each expert system tool has its limitations and is aimed at specific types of problems. Some allow the domain expert to quickly build rules, while others are programming tools that offer special integration features at the cost of steep learning curves and difficult rule development interfaces. For appropriate selection take into account that the project will be built, integrated and maintained.

Know the Tool and its Limitations

When starting an on-line advice project, assess if the system can be built with tools and techniques already familiar to the development team. Learning a new tool is time consuming and most sophisticated tools are not mastered quickly. Don't push the tools beyond their limits both individually and as an integrated project.

Many on-line advice projects fail because the developers go beyond the effective limits of a technology and use various tricks to extend the capability that the vendor does not support. Interfaces implemented in a non-standard way, are very difficult to maintain or enhance. When a technology limit is reached, it is time to consider if implementing extended capabilities are critical. If it is, then is the fix an acceptable solution for the projects entire life?

An outside knowledge engineer is often effective at bringing in a different perspective and wider experience to determine if there is a better, or a more appropriate tool. Often a desired system capability, that will have only a minor effect on the overall system success, has side effects or imposes requirements that drive the design and architecture of a project beyond the breaking point. If a project is running into difficulty during development, it is time to reevaluate both the technology being used and the project priorities. Focus on the key

Continued on page 49

Online Expert System Success Factors

Problem that can be solved with an expert system
- Expert exists
- Solution analytic, not based on intuition
- Defined limits
- Solution is understood and can be taught

Tool selection
- Appropriate to problem
- Expert can build rules
- Clean integration into IS structure

Interface
- Simple questions appropriate to the end user
- Vendor supported interfaces
- Key functions drive architecture

Table: 1

Hewlett Packard's CAST System – A Balance of Sophisticated Logic and Innovative Graphics

Hewlett Packard's *CAST/BW* (*http://cast.hp.com*) incorporates an interactive advisory expert system to help clients make purchasing decisions about HP products and services. The system provides quick, accurate hardware sizing, network configuration, and usage recommendations for SAP Business Information Warehouse (BW) implementations.

The knowledge engineers codified expert knowledge from SAP, HP internal competency centers, the HP Enterprise Server Group and existing SAP BW implementations. The system's user interface functions in a similar manner, as working directly with the company's most knowledgeable system analyst and product representative (Figure 1). The system results are presented as a printable HTML page complete with product images, system recommendations and configurations, and offers direct links to order processing (Figure 2).

The CAST/BW system is implemented as part of the Enterprise Systems Products Store which also offers product information, pricing and ordering of HP 9000 Enterprise Servers, VISUALIZE Workstations and Storage products. The sizing and configuration rules are continuously updated and refined to incorporate new research and new HP products.

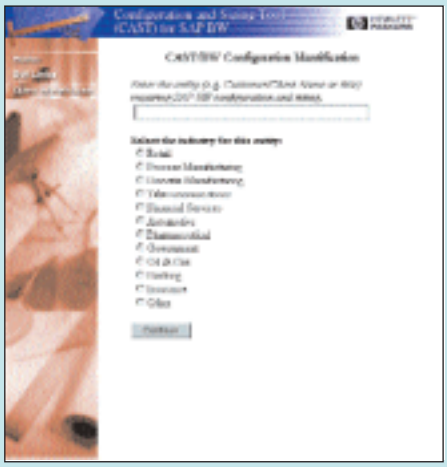

Figure 1: Typical HP CAST system question - easy to read and understand.

Customers are taken through an interactive query session that asks for pertinent details on current SAP R/3 environment business warehouse parameters, their industry, maximum number of concurrent users, number of "InfoCubes" (multi-dimensional data stores of individual business points of view) to be implemented, and whether cost or performance is the driving factor in the configuration.

The inference engine determines the best hardware configuration based on the rules in the knowledge base as well as the customer requirements. It then recommends the configuration, and provides a link to the HP e-Commerce Web page, allowing the customer to price and order the system. The expert system makes needed external calls to databases and data sources. The results page is dated, customer input is displayed, and a visual diagram with product photos show the appropriate equipment and system configuration, and details on processors and memory. The customer is warned of any problems in performance if significant upgrades are recommended.

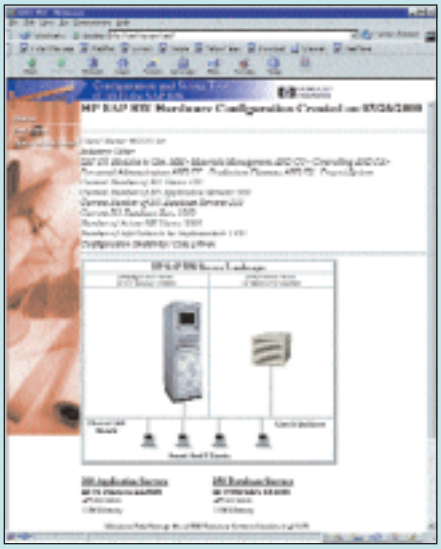

Figure 2: HP CAST system results – a recommended product configuration and dynamically constructed graphics.

Continued from page27
project priorities and don't let minor features drive system architecture.

Complexity of the Logic

There are many ways to describe and code the logic of a decision-making process, simple trees, and sequential If/Then statements. Some require the backward chaining capability of a sophisticated inference engine.

Determining the complexity of logic required to solve a problem can be difficult. A seemingly simple problem may have complexities that are not immediately apparent; or a problem that has high value to the company, may be quite simple to describe logically. An experienced knowledge engineer can determine how the logic can be described and processed. A little time spent in making this determination can greatly effect the outcome and success of the project. If a problem is simple and does not require an inference engine, it may be better to use C++ or Java to handle the logic. This approach gives the highest level of control and freedom of interface. However, this approach could be difficult to develop and maintain if the problem is too complex — complex logic usually requires an inference engine.

Resource Intensive Part of the Project

An online expert system project usually has two major phases – codifying the decision-making logic, and interfacing the inference engine into the site. There are many aspects to knowledge engineering, design and integration, and it is necessary to determine which parts are the most resource intensive both in development and in maintenance. Questions to ask include:

Is a human expert expected to build and maintain the logic?
Will the expert(s) be available?
Is the logic simple, but site design and integration complex?
Do the available tools meet the needs?

A tool loaded with integration features won't work if it is too complex for the domain expert to build the logic. Many inference engines are designed to be in control and this may effect other aspects of the project. If other programs will be controlling the flow of execution, make sure that the inference engine can function appropriately in that architecture.

Realistic and Efficient User Interface

Probably more online advice projects fail (or go over cost) due to user interface issues than any other reason. Since the system is part of the company Web site, the group defining and controlling site design should be involved. Even though the expert system may be considered as a show piece part of the site, it does not have to have complex graphics or a unique user interface. Combining two state-of-the-art technologies vastly increases the likelihood of failure. If the primary goal is to deliver knowledge that the customer needs and desires to help make decisions, do so through an interface that is easy to implement with the inference engine. The aim should be to emulate an interaction with a human expert – focused questions that lead to a conclusion or recommendation. Keep the questions simple, straightforward, easy to understand and easy to answer.

The better Web enabled expert system tools provide a high degree of control on user interface design either through HTML or Java. Keep the interface clean and simple. A complex graphical effect may be possible on its own, but may be quite difficult to use coupled with an expert system. If the goal is to deliver advice, you want the user to go away feeling that they have never before seen a site that gave such useful information. If advanced graphical effects are desired, put them either before or after the expert system, not integrated in it. This will also speed the interaction and put less of a strain on system resources.

Try to stay within what the tool easily supports. In general, tools that allow the greatest control are also more complex (more of a programmer tool) and are more difficult to use. This means a significant increase in the effort to build the expert system logic itself. Make sure that the benefits are worth the extra effort.

A Success Story

Despite the potential pitfalls of adding online advice, it has never been easier. New tools make it possible to describe decision-making logic through easy to use visual interfaces. Developed systems can be delivered via inference engines either running on the server or run client-side as Java applets.

Hewlett Packard has done a great job at showing how a powerful expert system can be integrated with a very sophisticated user interface (See side bar). Their interactive expert system is a good example of using the right approach to a problem, appropriate developer tools, and an effective user interface. You can visit the HP site and run the system at *http://cast.hp.com.*

Thinking it Through

A properly thought-out and engineered expert system project provides tremendous payback. HP's *CAST* system reduces days of work, phone calls, and emails into a few-minute online interaction that produces an expert recommendation. Selection and configuration of which products best meet a customer needs and requirements can be a very intricate and complex process. But it is one that can be expressed in logic rules relating to customer needs and product specifications.

Advisory products built using expert systems can handle conflicting requirements and always deliver a recommendation of the best fit, even when all customer desires can not be met. Real-time data, such as inventory, current pricing and customer requirements can be included. Some expert systems also enable change and rerun as well as testing different configurations based on different criteria — a cost driven vs. performance driven comparison. They also make it possible for staff to identify cross-selling opportunities and be able to sell a much broader, more complex product line.

The Internet is the first line of communication to customers. That fact will require vendors, as well as customers, to become smarter about how best to use the Internet infrastructure. Just providing information and graphics on Web sites isn't enough to give businesses a competitive edge any longer. By incorporating interactive advisory systems, company's can offer expert advice. It's like entering an online store — you consult with their best sales person who helps you select the best product for your specific situation, without any personal biases on the part of the sales person. Online advice will become the next standard for the Web.

Dustin Huntington is President and founder of EXSYS Inc (*www.exsys.com*), a company founded in 1983 for the development of expert system tools and applications. EXSYS Inc, one of the first companies to design expert system development tools aimed at the domain experts, was also the first to offer a Web enabled inference engine. He is the lead programmer of the EXSYS Developer and Web Runtime software and, soon to be released, Corvid' expert system tools. He can be reached at (505) 281-9400 or *dhuntington@exsys.com.*