By David Holmgren, Senior Financial Analyst at Ivory Consulting Corporation

Beyond the cliché that the Internet is transforming the way business is conducted across industries, let us examine the implications in one specialty: lease analysis software.

Lessors, lessees, brokers, and other financiers use such software to structure or price leases and to analyze their financial effects and profitability. The points discerned here will probably pertain generally to software used by the leasing industry.

Pricing software has its genesis on mainframes in the 1970's during the formative years of leveraged leasing. Such mainframes were maintained by large financial institutions for their own structuring use, or accessed on a timesharing basis and maintained by staffs of dedicated professionals. During the 1980's personal computers became widespread, along with new software programs, which grew in capabilities and features. During the 1990's the use of these programs was fueled by the growing popularity of Microsoft Windows and the rapid increase in the speed of personal computers. Timesharing-based services continued to have a significant presence through this period.

Today, the Internet is producing the scenario of centralized financial software accessible through personal computers. Interestingly, this is reminiscent of the timesharing paradigm of the 1970's, the main

difference being in access over the Internet rather than a direct dial-up connection.

Another trend that has exerted an influence on the design of analysis tools during this time is customization to meet specific organizations' needs. Competition has forced companies to build on their strengths, which often means more carefully measuring and allocating transaction costs, in hopes of gaining a marketplace edge by making deals more attractive while still meeting profitability targets.

Customization facilitates consistent financial metrics and analysis of company-specific information on funding and other costs, credit, equipment, risk, and taxes. Further, some marketplace segments have moved toward more complex structures, motivated by creative interpretations of tax and accounting procedures.

Making a deal more "attractive" means structuring for different objectives, depending on what a particular customer is focused on. One customer may be interested in the PV of rents, another in the all-in run rate, while still another in the capital/operating distinction, to name a few examples. Thus, lease bids are more than just bids---they are the strategic tools that cover an organization's costs and risks while meeting multiple structural requirements. As such, they contain complex, sometimes proprietary information that cannot be handled without specialized or customized tools.

Much customization is in algorithms and calculations (the "engine"), to be sure, but a great deal is also in the interface. Indeed, sometimes the interface is customized more than the engine. A major benefit of a graphical user interface (such as Windows) is its "intuitiveness," meaning that the presentation of information itself guides the user in using the information.

Let's say that we are structuring a 5-year lease with a fee. The amortization of the fee depends on a method and a term, among other things. Assume that after specifying straight-line with a fixed term and a 7-year term, we switch the method to straight-line over the lease term. With this method, the amortization term is not required, as it is set directly from the lease term (5 years). The program will "disable" the amortization term, meaning that its display is dimmed and made inaccessible, providing an instant, intuitive signpost about the source or use of this information. The processing to support this treatment is in the interface, not the engine.

With a stand-alone program, the interface and the engine exist together, but a web-based application splits what formerly was one program into two: an interface on the workstation and an engine at the website. Such a configuration may disappoint users who have become accustomed to the crisp and responsive performance of stand-alone programs. If each user action and response has to be shipped across the Internet, even the simple disabling of a field can involve a noticeable delay.

Anyone Can Use It

The beauty of an internet-based pricing tool is that anyone can use it through a browser, rather than installing the actual program on their own computer. The hardware and operating system choices are also irrelevant, other than to support the browser. Perhaps the most important link is the speed and reliability of the link to the Internet. Thus, with a browser, a user can access the full capabilities designed into the website with no other software. The developers of browsers have the resources to make their products reliable, robust, and flexible to interact with a variety of hardware and software platforms. The main disadvantages are that they are relatively slow in processing and less powerful than a modern object-oriented language.

A number of advantages of a web-based service may outweigh some of these difficulties. First, all users would have access to the latest version. This is especially desirable when centralized tables are used to drive calculations. Transfer of files between users will be straightforward. By extension, transactions can be easily aggregated for management and analysis. It also facilitates user assistance, including diagnosis of problem cases.

Perhaps the most widely touted long-term benefit of web-based applications is the promise of integration. For example, information collected during the sales process is transferred directly into the pricing process; final pricing numbers are transferred to accounting for booking the lease and subsequently tracking its activity. The web can

facilitate this strategy by presenting a convenient electronic interface for the applications.

As it turns out, web-based applications are not required for system integration; it can be achieved just as well with stand-alone programs. What it does require is open program designs that allow data interchange, i.e., the applications must receive and send data according to common formats.

Usage Charges

Usage charges for website processing are an area of continuing uncertainty, and it may be some time before a market consensus emerges. In contrast to the service of business-to-business ("B2B") sites that charge fees based on equipment sales that they facilitate, a pricing-website does not have a correspondingly simple basis for charges. Its costs are most closely tied to developing and maintaining the service. At the same time, most personal computer users are accustomed to a fixed charge per user, which does not lend itself well to a service that would seek to make itself available to all users, existing and new, regardless of their usage. A web-priced service is more analogous to a timesharing service where charges are based on actual processing. A fixed charge per user is certainly also possible, but that works best for established regular users, not occasional users. The distinction between accessing a website and loading a program on one's own computer may become insignificant, however, as disk space becomes less costly (practically free, once a user has a

computer). Further, providing applications through email or a website has become standard procedure, making stand-alone installations quick and convenient. Associated files --- such as rate tables, templates, and reports --- can also be easily distributed via email, so that keeping users on a uniform version is not as difficult as it once was.

The distinction between accessing a website and loading a program on one's own computer may become insignificant, at least with "well-packaged" programs. Disk space will continue to become less costly (practically free, once a user has a computer). Further, providing applications through email or a website has become standard procedure, making stand-alone installations quick and convenient. Associated files --- such as rate tables, templates, and reports --- can also be easily distributed via email, so that keeping users on a uniform version is not as difficult as it once was. However, updating a program that involves several components is not straightforward and must not be trivialized. The specifics of the programs in question should be carefully considered.

Quality control is one of the more vexing difficulties facing software developers, and it affects all users to some degree regardless of platform. Ideally, testing calls for all combinations of features to be exercised, but the complexity of an elaborate financial model makes that impractical. Any piece of software that is continually evolving is even more problematic because existing tests need to be updated to

stay current with new functionalities, and new tests need to be devised to exercise new features. Regression testing should be a part of any developer's test procedures. This testing compares values produced by a new version of a program to values produced in the past (and which have been accepted as correct) to determine whether the integrity of past results is maintained. Differences need to be examined carefully and the benchmarks kept current with updates of the program. The more comprehensive these tests, the greater the assurance that the program maintains the integrity of past results.

Challenges to Quality

Web-based applications present greater quality challenges than stand-alone programs. Each additional software layer or access link is a potential source of bugs. Furthermore, the additional levels of interaction can trigger problems that otherwise would be benign or unnoticed. With web-based applications, pinpointing bugs is more difficult, as it is not necessarily obvious to the end user where something in the long chain of processing has gone awry. Indeed, sometimes just reproducing the problem is a hurdle (let alone diagnosing it), inasmuch as the environment is so dynamic. Yet, problems anywhere in the chain can prevent the end user from receiving their desired result and must be guarded against.

The convenience and reliability of Internet solutions will remain issues for some time. A centralized web-based pricing tool may be a plus, but it still requires a hardware connection (at least until the typical laptop

comes equipped with a high-bandwidth cellular phone links). Do the Web advantages outweigh the strengths of a simple stand-alone application with its response, security and control?

Slowdowns and outages of Internet access are another reality that should not be minimized. When a deal needs to be priced, time is of the essence. Public availability entails security risks, either through malicious intruders or simply additional exposure to a larger audience with a correspondingly larger variety of ways of using the program. When transferring sensitive business data over the Internet, an encryption scheme should be used. However, moving to web-based pricing probably does not in itself significantly increase a company's exposure to Internet risks, because it is likely already exposed through its email and website links. A number of approaches can be employed to deal with security risks, but that discussion goes beyond the scope of this article.

Pricing software will be represented in several operating paradigms in the coming years. Analysts who do extensive or frequent pricing may prefer to have the program on their own computer or to rely on timesharing services accessed through direct-dial connections. Other users, especially occasional ones who do not require customized software, may prefer to use a browser to access a web-based service and not concern themselves with installing and maintaining the program on their own computer.

Companies seeking the benefits of a customized solution will have the choice of a stand-alone or a web-based installation, and several points will factor into this decision, including choice of features, use of centralized data, frequency of updates, and eBusiness strategies. For many companies, system integration will likely be a major objective in moving to a new platform. Technological progress does not necessarily promote one paradigm to the exclusion of others; it may help all of them, but to varying extents and in different ways.

By David Holmgren (Published July 2000 in Leader's Equipment Leasing Newsletter)