BackOffice and the Internet

Transcript of Todd Warren's presentation at the Professional Developers Conference.

March 13, 1996

Todd Warren
Microsoft Corporation

Download the Microsoft Word (.doc) format of this document (zipped, xxK).

Editor's Note: In a few cases, we have added in words that were apparently dropped in the transcription process. They are set off by brackets: [ ].

TODD WARREN: It's great to be in San Francisco. I'm a group program manager for Microsoft BackOffice. I've been here nine years, so it's always exciting to come to talk to you about a show.

On a beautiful day like today, the San Francisco bridge and in many ways bridges in general are a good analogy what we're doing with BackOffice. It's a beautiful piece of architecturing of itself. It provides incredible function for people and communities of interest. That's what we're trying to do with BackOffice.

What I'm going to talk about today is BackOffice to get you started and take you through each of the pieces of BackOffice and tell you how you can use those to better enhance your Internet experience. The breakouts this afternoon will go into more detail into all the pieces I'll cover. Then what I want to do is show you two applications for shipping and production applications that bring it all together that show you how you need to use these different pieces in tandem to produce a better, more active Web Site.

The last thing I will talk to you about is a little bit about how we're bringing things into the future with BackOffice. Some of the additions that we're adding to the base frame structure and file system, base frame structure in all the various applications to make it possible for you to build even more powerful applications.

Microsoft BackOffice today is a set of four server applications built on the solid foundation of Windows NT™ server. Basically those applications share a common user interface in terms of administration, so it's very easy for an administrator to manage a set of services distributed on various servers. We provide a number of common pieces, performance monitor, event log that make it easy to track problems across the different service. We think it's the easiest server environment to manage for multiple server applications. We have a common security model that is for security. All of our applications take advantage of the single-user log on provided by the Windows. Leveraging this in your Internet applications is more possible for you to build both great intranet applications that are secure and give people the right access to function, as well as Internet applications that are secure and robust.

The networking model of each of the BackOffice applications is designed to work in a protocol, independent way. This is good for building Internet applications and train net applications. Every one of the applications works in both IPX and IP environments and can have a mix of the two in terms of building together and building your applications. We also provide a good set of programming devices and tools to build those applications. Each one of the applications is available on the wide variety of hardware platforms that are available on the market today, including Intel and Intel SNT systems, as well as Alpha chips and power PC. In the future we'll be expanding the set of services provided as part of BackOffice. Today we have Microsoft Mail

and very soon Microsoft Exchange, as well as SNA server, SQL server and Systems Management server helping to manage desktops and servers in the network. In the future we'll be adding an Internet PROC server to make it easy for companies to access information in a secure fashion. We'll also be integrating in the merchant server. Bill showed it in his talk as part of BackOffice, as well as in the future adding the on-line transaction processing technology that we call Viper, as well as the video server or media server we call Bengal.

All of these are built on the same common Windows NT infrastructure and integrated with the Internet information server that's provided as a base part of the Windows NT system. Each of these can be leveraged by both your applications, by standard development tools and of course by our own applications, including office, the authoring tools for the Internet, Front Page, Internet Studio.

One of the great things about BackOffice is it's a super exciting way to build intranet applications. In particular, as both Paul Maritz and Bill Gates talked about earlier in the conference we're working to take all the different kinds of information that exist within an organization, all these multicolored threads and weaved them together into a single tapestry one can view across the organization. Not all the information is in a static HTML home page on a server. We do this of course for Web files we do it for messages for the exchange server. We'll talk about some of the things we're doing there. We also talk about some of the additions we've done to SNA server that will let you do it easily for host data as well, specifically with the DRDA driver we've provided.

It's very easy to host database information on AS-400 or DB-2 system. SQL server for structure data makes it easy to build an active Web site that's generating information on the fly based on user queries and allows you to build a site that is customized to a particular application. All of course building on the common infrastructure of Windows NT and leveraging your investment in existing files and existing file formats. Overall for the BackOffice we really see it as the core of the Internet, going beyond what you can do with static Web pages, scanning the full spectrum of data storage, address things on the file system, integrate them with messages and structure discussions that are part of the exchange server, getting out mission critical data stored in SQL and making that available reliably and safely, making it easy to distribute around the organization a wide area environment with the application functionality in SQL server and Microsoft Exchange so you can easily access it, providing a single integrated security model so you don't have to manage separate password lists and user IDs on the Internet for applications that use data from a variety of locations and having a common set of administrative tools for bringing this all together.

We're moving beyond this as we move into future versions of the different pieces of the BackOffice. Today all of our administration has a common look and feel across the different applications. In the future you will be able to build server applications that have a single administrative interface across multiple different server applications. We're extending what we're doing in security to go beyond the NT challenge response, security mechanism, an authentication method with rows and T servers.

We are extending replication beyond the exchange server and SQL server and extending that to the file system, will enhance our performance and reliability for data storage and make it possible to do more distributed kinds of data access and on-line transactions. Our host integration will continue to get better and integrated with the security model. The other thing that's really been at the core of Microsoft's mission before the sea change of the Internet is [to] make it much easier for people to access information, make it easier to do information query. We're going to make sure that it's easy to do this both across HTML files as well as the existing files that people have in their system, and across the data that's in your applications by providing open interfaces to make it easy to query that data. Of course, all this leverages with

Windows NT a common NDI desktop that makes it easy with DCOM to distribute the load between the client and server.

What I want to show you is a new piece of Windows NT technology we're working on. This new piece of technology makes it possible to serve HTML files that are available on a server as well as arbitrary data formats. We provide a set of filters to get all the existing data you would expect and we support an interface called I-filter so you can write an application that publishes its information so it can be content index. I want to say the ActiveX search technology will be a common part of the Windows NT platform and available in Beta form this summer so you can try it out and start writing the filters that are specific to your applications. We'll provide a standard set of filters for the common file types that are available in the organization.

Let's go to the demo. Here what I have is a especially built version of www.microsoft.com. This page is available in the hands-on center if any of you want to try it out. We've taken www.Microsoft.com, and using the search and technology we have built all the HTML files, all the doc files, all the PowerPoint® presentations that are available on top of the system. This happens automatically in the background. In fact, we can do, on a raw system we can do about a gigabyte of information in an hour with the contending index technology we're building in ActiveX search.

We provide broader ways you can query this information so, for example, you can use just a free tech search where you type in some information. You can use bouillon and proximity operators so you can look for information near another piece. You can use wild cards and free text. You can also do vector space type queries so you can easily find different proximities of particular phrases. We also do things like stemming so that if you're searching for a fly, we'll find things like flight and flying and other derivatives of the particular phrase in the system.

So let's go ahead and do a search on Microsoft.com. Let's look for Bill Gates. We really want to look at those documents that are near Internet strategy. What happens here is the searcher will go out and get a listing of all the documents that meet that particular criteria. Here you can see it match 36 different documents where Bill Gates shows the Internet strategy and it shows me the first ten. By default we have the setup to show ten at a time so I can skip through them. You see the first one it found was a document that made it easy to look at the support of the Internet and Windows 95, as well as Bill Gates' strategy presentation. It did take into account the data that was in his PowerPoint presentation, as well as the data in the HTML transcript. We can formulate and rank that information as we searched it within the system.

Let me try a different kind of search. Since I'm very involved in administrative tools at Microsoft they have been very interested in job scheduling. Let's say I wanted to find out about job scheduling in Microsoft applications. It could be under job schedule, it could be under a bunch of different derivatives. I can type a wild car search, execute query and it will bring back a set of applications or a set of documents that have to do with job scheduling. You see in particular it highlights the different pieces of SQL server 6.5 and job schedule that's available. I see things like this file that talks about job scheduling and so forth.

One of the additional added values that we provide as part of ActiveX search is not free tech search HTML documents and existing documents in the system but the ability to query by property on a particular document that's stored on an HTML server or any of the servers linked to that server via the route capability of the server. If I click on the document author document query I have the capability of querying on the standard summary information doc file property associated with author. So here I'll search for Collins and Smith and execute a search. Well, I didn't type it right. Let's try that again. I meant to say "or there." Here I get the set of documents that were authored by anybody named Collins or anybody named Smith. You will notice this didn't pull back HTML files, it pulled back all the Word documents on [www.]microsoft.com authored by someone named Collins or Smith. I have the opportunity to

open those directly as part of the hits that were pulled back. So there is an example of the kind of thing we're doing to build on the basic infrastructure and platform to make it easier to find information and build more extensive intranets as part of your [application].

Let's go back through each of the pieces of the Microsoft product line. That's a lot of how we're enhancing the foundation of Windows NT. We really see this as the main server application foundation that we will continue to extend now and into the future. The key attributes of Windows NT Server are high performance and scalability. We continue both with our platform support, with the tuning that we do in SMP environments, with the work we're doing in clustering to make it more high performance and more scalable. We're trying to provide the best integration into the Internet. One of the key points here of course is the work that we've done in terms of multiprotocol support. Since we know of TCP/IP, we support protocol independence in terms of application writers building things both with TCP and DCOM and window sockets, as well as providing easier ways for people to roll out TCP/IP with the DNTP and WinServer. We also with Windows NT 4.30 indicate with DNS to provide dynamic capability to register machines as part of the DNS.

We really have worked hard to provide the best server application development environment, both in terms of providing a core set of administrative tools you can extend, providing good security services, architecture and high-peformance I/O model available with things like completion port and Windows intake.

How can you build a great server applications on top of Windows NT? There are a couple of core things that we ask applications to do that you have probably heard some about this week. The first is making sure you run as a service and use Win32 to get high performance I/O. This is how we're able to get the high levels of scalability like intranet information server. As I think was said in the movie Animal House, "Fat and slow is no way to go through life." Fat and slow is no way to build a server application, so we really have given you a rich environment which to build that server application to tune it and let it scale as you add more hardware to the environment.

We've made it easy for you to integrate into the Windows NT security model. This information server lets you set permissions on a particular file and on a intranet deny access to those particular pieces. It also lets you, for example, when you are using the Internet database, connect or have a log onto the database accessed through the database connector and to the server and the network itself. We added APIs and NT 3.51 log on user to make it easier to make an application that provides a single ID. SQL service 6.5 is the high performance database for structure data within the system both in terms of providing the best price performance ratio as well as optimizing for ODBC access.

Of course the side effect of that is that SQL server ends up being a very good environment to build dynamic Web applications that use the Internet database connect or included in the Internet information server. We've added support in 6.5 to support heterogenous replication, so when you couple it with some of the other elements of BackOffice like the SNA server you can stage data from a SQL database up to the host using the heterogenous ODBC feature of SQL server. If you are building a series of servers that have common data and are using the Internet data, you can distribute that information using the heterogenous replication. We've also added some additional things to make it easy to take data that's in a SQL server and automatically generate HTML for a Web Site. So, for example, in SQL server 6.5 we include an Internet publishing Wizard that makes it possible for you to walk through selected series of tables, or select a stored procedure or query, and either immediately or on a periodic basis, or as a result of a trigger, have that HTML page generated.

So, for example, if you think of, say, a Web site of product information and one of the things you want to have as part of that product information Web site is, say, a page that has all the

items that are out of stock. As soon as the inventory of a particular item goes to zero, a trigger could be fired that could regenerate the out-of-stock HTML page for people to access. So we're really taking SQL server and making it much easier for people building Web based applications.

The other main piece, of course, is the Internet database connector that both lets you get at SQL data as well as other ODBC information. The way this works is really by providing two simple files in connecting to the SQL server. The first is an IDC file that provides the essential information to ODBC to connect to the particular database which stored procedure to file once the URL is accessed. Then once that query result comes back, you create another file on the Internet information server called an HTX, file which is a template for how that information should appear. The HTX format has some pieces to help you specify where the detail goes and where different branching should occur based on the results of that stored procedure or the content that was queried via the database connector.

We have a breakout session where we're later going to walk through a real Word, pretty extensive example of using this to bring back multiple result sets and leverage an extensive procedure on the SQL Server. This is a product I'm excited about. It's a substantial addition to what we're trying to accomplish to the BackOffice. That's reliable client server messaging with Groupware functionality. We'll ship it in the next couple of weeks. It's close to going golden and it has a couple of awesome features to help you integrate with the Internet.

The first thing included in Exchange Server 40 is SNTP connectivity built into the system. You get the mail connector as a standard part of the server. This supports mind message types, it supports MAPI of the transfer, the neutral and capsulation format, or the Net to send exchange forms over the Internet to other exchange applications or client users on the other side. It provides all the functionality that you would expect in a high performance, client server messaging environment, including a very rich information store that indexes messages, provides replicated folders for discussion and for shared applications.

We're also working on a couple of additions for the next version of exchange that enhances Internet support. We're covering some of those at some of the breakout sessions later today. In particular what we've done is something called the Web connector that allows you to publish information in publish folders live to an HTML file. That also includes the ability to interact with those public folders so via HTML form someone can respond to a question on a discussion group and post that information directly to the public folder. We're also working on an NNTP connector to public folders. We have this running on Microsoft. We're taking the full feed off UseNet and Realtime and putting that in folders. At the time it's pumped into the server it can leverage exchange replication to replicate that around the network and give people access to that information. Also, of course, because it's in the Exchange store you have the capability to do all the kinds of indexing that are available there to create views and and track conversation threads as part of the system.

The last thing we're doing, and it's important for other Internet clients to access Access Exchange, is adding support for POP 3 to the exchange server so that if you have a POP 3 client either the exchange client with the POP service provider that's provided in the plus pack or Udora, you can use that front end to access information in an exchange server. I want to talk a little bit about the Web connector and how it's constructed. We built it as an ISAPI application with an extension called Web dll.exe. Once you give the URL to connect to the Web connector you basically specify the folder, the public folder you want to access as part of Microsoft Exchange. The Web connector will call MAPI, the MAPI gateway APIs and MAPI store APIs to extract that information from exchange and publish it out to the Internet as HTML. We then for the standard post forms included as part of the exchange, provide an HTML form so you can post that information back to the system. All this is integrated with the security that's provided

as part of the Internet information server so people can have secure access to that information that's published in public folders.

We're also working on the news connector. Basically the way this works is there is a service that runs, that takes the NNTP feed from the Internet, goes through the connector, calls the MAPI, store APIs to write to the server and create the public folder hierarchy to create the MAPIs for the users as people respond and post information to NNTP. And then because the information is in the public folder store it can go around to the individual servers that are available. There are a lot of ways you can use MAPI as part of your ActiveX™ server or client application. The first is to use the message store-in. So if you are building an application that runs on the server that needs a level of indexing over heterogenous state and not highly structured data as in SQL server, you can use the exchange information store as a way to do that, especially if you need high availability across multiple servers, the replicated storage is a good place to put that. It's also useful if you want to have data available both to people using the exchange client and creating views, and doing filters, and doing searches with that or information in a more unstructured or Web client environment. You can put the information there and both kinds of clients can see it with the Web connector.

With client applications, if you have an application that wants to take advantage of sending mail on the Internet, the easiest way to do that is with simple MAPI. Because we provide both in Windows 95 and the PlusPack and standard part of Windows NT 4.0, the POP 3 connector to the Internet, you can call MAPI send mail and that information can go out over the Internet. That of course also works well with any of the other transports that have been provided as part of the MAPI environment. So of course it gets you to Exchange, gets you to Microsoft Mail, CompuServe and so forth. You can also, when you attach messages via MAPI, send and specify a particular mind MAPI site for information to go out as if you are operating with other Internet services and need to have a specific mind type. You can do that with Internet Mail connector or the provider included as part of Windows 95 and Windows NT.

Custom message types or E-forms also have the capability to be sent over the Internet. In fact, I'll be passing out today at my breakout session a new version of the transport neutral and capsulation format which describes exactly how these MAPI properties and MAPI information is transmitted across the Internet, so if you need to impact it on the other side from a nonMAPI client or nonexchange environment.

SNA server and SNS provide good ways for the Internet, especially for managing servers on the Internet, and for unlocking the data that's in the host. With SNA server we provide ODC driver to ODBA that makes it DB-2 data that's a standard part of SNA server. We provide FTP to AFTP gateway so you can download files from the host to the system. This fits in with the basic infrastructure with the server. You can also, if you are building a direct host application and you are building an ActiveX type of application, you can access the host via IP over the Internet with the SNA server acting as the proxy to use the SNA protocols to get back to the host. This makes it possible to bring layers of applications into this new easier view on all the information that's available. SNS makes it much easier to manage Web forms.

One of the things we'll cover in the breakout is how to distribute an ISAPI DLL as part of SNS's software distribution. Also, with some of the capabilities we're in the process of adding SNMP alerting, if it's very well into a network or for managing multiple servers for things like event log coalescing. We'll also be adding to the BackOffice the merchant server. You saw a little bit of that in Bill's demonstration this morning. That includes client APIs for doing shopping and you saw a little of that in the demo, including the facilities to doing encryption payment, the ActiveX client controls for doing local order caching and for keeping track of address information for a particular user. And on the server side providing the links to all the data that's in the organization.

Again, one of the key points we're trying to do at BackOffice is take all the data available and bring it back for availability. Billing systems are part of this. Michael Cockrill drilled down on this more in his breakout, but basically at every level of the system we provided a way where you can escape and get to the data that's specific to an organization. So in particular there are APIs to make it possible to get out and get at product databases that may already exist in the organization, to get out other payment systems that may be part of the system and also to get at legacy information that's in accounting systems and other information systems within the organization. With that, that's really kind of a broad-brush overview of the particular APIs and the particular servers. I encourage you to go to the breakouts on the Web connector and on the different pieces of the BackOffice. We're showing you how to do some of those particular server applications.

What I want to do now is bring out Mark Tebbe, president of Lante, and take with you the Lante site system. The Lante on-site system, which a lot of you have been using, actually has been getting quite a beating as the activity center for you. It's implemented on top of the Internet Explorer and Microsoft BackOffice. It uses SQL server, Exchange and the Internet Information server. Let's introduce Mark Tebbe, president of Lante.

MARK TEBBE: To build a system like this, and it's the first time it's been done on an Internet, you see people lining up and using it to run the daily business. We just randomly set mailboxes to be maximum of 100 megabytes. What we saw was people have exceeded it. To run through 100 megabytes of data, we're watching these people and they are running the business having all the mail forwarded from their offices, forwarded here. We're also seeing a lot of people spamming things out.

TODD WARREN: There is so much spamming it's a regular Hormel plant out there. Why don't you show me a little bit of the system?

MARK TEBBE: What we have here is the ability, when you walk up to the kiosk to do an NT logon and you will be presented with this screen which says "Welcome" to you and gives you a little bit of an overview of what we have to offer.

TODD WARREN: You take the NT logon information and customize it to the page.

MARK TEBBE: We point to the home directory and update the registry with all the information so we know exactly who they are and what the schedule is.

TODD WARREN: You probably fill in the name here and the rest of this is a static page because I imagine it's the same for everybody.

MARK TEBBE: No, matter of fact what is cool, this entire system is built with dynamic HTML. There is no static HTML in the entire application. What we have is the news of the day. We could customize per person if we so desired, same thing with the menu on the right. Some people who use it, like Chris and "J" Allard. From here the person has the ability to go in and see information about the conference, scheduling our agenda and the communication network itself. This information is actually the same information that we have at www.microsoft.com on showcase event. I can come into conference information here entirely built dynamically. I can see facts, maps. Let's look for competitor information. From here I say I want to see a map or search. I pick a search. Here I can fill in the search which is, again, all dynamic. I fill in the search, short description of what they have, a booth number or search it alphabetically. Here are all the exhibitors that are at this show that start with the letter C. I'll pick Citrix. Here is the information about Citrix. If you didn't know anything about that and you wanted more information, we have it here. What you will notice is when we got the information as an exhibitor they identified they did have a home page. You will see this Citrix name here is

actually a dynamic link, again, entirely pulled out of the SQL database. We have something that points to the home page. Click here, go out to the Internet and pull up Citrix on page life.

TODD WARREN: If you needed to add a new vendor how would you do that, if vendors were added?

MARK TEBBE: Our role is to develop the applications rather than maintain them. We want to show managers to do them. We have an access to literally enter a SQL server to insert a row. If we added a new exhibitor here we put Citrix system in, the address, the description, the Web page the booth number and the like. You will see also the booth number is dynamic. What we have is the ability to see a booth number, pull up the exhibit hall and show exactly where they are.

TODD WARREN: So it's easy to find the information and easy for someone to manage the site by adding the information to the SQL base?

MARK TEBBE: Right, the person doesn't have to know that much about it.

TODD WARREN: How do you use the personalization features?

MARK TEBBE: Once we have the information and know who everybody is, we are able to maintain the person's schedule. If you look here at my personal schedule for today I literally can switch to any other days there if I so desired. I see here that on, item 9, which Bill spoke of this morning. And I have a discussion link that goes out to exchange and I could go into a threaded conversation regarding what people thought of it, what they thought of his video and the like, just right there dynamically.

TODD WARREN: You don't need to do that. We have a breakout session where we're going to go through exactly how each of the pieces of this was corrected, database schema how the links were set up to the public folders, how the stored procedure and Web connector was built.

MARK TEBBE: That's true. What you will see here is in my personal schedule. I have added this particular session called pouring the Internet with BackOffice which I was alluding to. Here I can click on it and pull up all the information itself and see it's today, and a short description of it. Todd is in charge. Then on the right-hand side, again, dynamically I see what track is related to it, the track layout. I see it's a doubled session. Jump out and point to exchange. I can send E-mail to Todd or if I'm going—you know what, if I'm not sure if I really want to see this, we have again a pointer that allows us to go out, pull out a copy of Todd's presentation and read it right here.

TODD WARREN: So how do you do that using the database connector? How do you set up your schema an the different things to do that?

MARK TEBBE: We have a pointer in the database that says Todd Warren's presentation is WARE, FFTPPT on a common directory. We launch that and bring it to the browser and launch the PowerPoint viewer to the browser and load the slide. So you have a link in the SQL base and that activities file in the file system.

MARK TEBBE: Yes. The way which we're running the demo here, this could be happening at www.mark.com to where if you wanted to, after the event, go up and run the exact thing you have that capability.

TODD WARREN: Tell me about the development cycle for this application. When did it start? I don't know if it ever finished. How did that work?

MARK TEBBE: It's finished when the show goes live. We finished about three months ago, [that] was when Microsoft said we want to see if we can build this on site. We pointed out there is a lot of opportunity and wanted to jump on an example. Can you do this? Can you get it up to www.Microsoft.com in two months? We said, "Yeah, we can do that" and we did. Then we took another month to build the local component. From start to finish it took three people about three months. From that we've gotten an idea of the final system. What you are running today is literally less than three months old.

TODD WARREN: That's great. Thanks a lot for a great system. We'll be having a breakout session where we go through in detail where the Lante system was constructed, combining the database schema and walking through how the database connector was used, how exchange was linked in as part of the system. It's used to recap. It's used for the registration of different people as part of the system. We use the Internet database connector. Every page is done with the Web connector including the log on, including all the menus and registrations as they come up. Then there is a SQL server that contains all the descriptions of the courses, the particular links to which sessions a particular user is interested in and so forth.

One of the nice ways that we're making it possible to expand that is you notice today, for instance, to link to an exchange public folder what Mark had to embed was a shortcut that activated the exchange client to view the folder with the exchange Web connector that Tom McCann is covering in his breakout session. You would be able to activate that via URL and SNAI in the environment. We're making it easier so viewing that structured information in SQL to some semi-structured information in exchange or to the unstructured information that's in the file system with things like the PowerPoint slides.

Now I want to talk about another application that's been developed by a company called Software Business Technologies. Now, SBT is a company that's been developing applications in the PC market for a long time. They both develop accounting applications and in the last year or so they started to develop Internet applications. They have lots of users and a big channel of bars that customize their accounting system for different customers. They initially based their system on FoxPro and have been evolving it to use things like SQL server and the Internet as they have added onto it. They have two main products, a Web series product that basically allows someone to take their product information that's inside their accounting system and publish that easily on the Internet, and then let people order information and have those orders go directly into the accounting system. They also aid small businesses in using this through a service they have called Web Street, which basically let's them host a series of home pages and product information so they can order the different pieces.   SQL is the application we'll show you in just a second.

Really, what it lets you do, it lets a small business take their product list that's already in their accounting and inventory system, take the payment system and the accounting system and build a full functioning presence in the Internet linked to the internal system. Take an additional client server application, and extending it to take advantage of the Internet. They have added intelligent ordering that makes it easy to let a customer order, let a customer electronically check on their orders as part of the system and they also provide integration with other order entry system if you are not using FTP's accounting system. They started the application before we released the Internet information server and implemented it on O'Reilly Web Site. After they moved Web Street, their Internet presence, to NT Server they saw a lot of improvements.

They saw a lot of improvements. They saw much better performance. Much easier integration and much easier database access and security. They've used, for example, the Web Trader Application, the Internet Database Connector.

Now what I'd like to do is bring out David Harris, who's president of SPT Internet Systems, to take us through the Web Trader Application. Thanks for being here, David. Yesterday you were

telling me a little bit about SPT and sort of how you approach some of the sea changes like graphical user interface and the Internet.

DAVID HARRIS: The accounting marketplace is conservative. These people think the DOS applications [are] the breakthrough technologies. We move them through Windows and we're moving them through the Internet.

TODD WARREN: How have the tools evolved?

DAVID HARRIS: We started on Unix boxes. We moved over the NT box and now we're using Internet information server to running all of our sites. We have about 50 domains.

TODD WARREN: That's great. Let's take a look.

DAVID HARRIS: We're looking right now at one of our Web street malls. This is a mall that shares product and services to end users. We're going to take aA look at category products up here that are displayed. We see that the New York smoked fish company has a home page.

TODD WARREN: Is that a real company?

DAVID HARRIS: That's a real company in New York. These folks are actually taking orders from the Internet. They are currently a wholesaler and they wanted to experiment with a retail presence. They found the Internet was an ideal way to share the products. The cost of opening up a retail store in the tri-state area is immense. This allowed them to solve that problem, test direct sales without having to open up a physical store. The catalog is being displayed on Internet Explorer. There's a little movie. It's nice for them. That movie shows automatically on the Explorer whereas other browsers require more. You take a look at the catalog. This is coming out of their inventory database. The items that are shown here are coming out of a SQL database. In fact, the IDC kit that comes with the information server to allow us to display the 23 information, so the information is current. It's out of the corporate database and they don't have to create extensive HTML catalogs from scratch like they used to.

TODD WARREN: That's great. Based on their inventory and the products stored in their accounting systems the Web site will give that data.

DAVID HARRIS: That's correct. If we click on something like the picture. We can go into the ordering processing cycle. We're not only showing a dynamic page here but actually a data entry session that we're having.

TODD WARREN: You can push the data entry out to their customers that do it themselves.

DAVID HARRIS: That's right. The banks did it 20 years ago when the ATMs were designed. I can go ahead and start this order entry process and the system went out and recognized that I was a prior customer. Through stored procedures my name and address is brought up here as well as a discount level based on the amount of smoked fish I ordered in the past.

TODD WARREN: It brings up a price list based on the information stored in the accounting system about the customer.

DAVID HARRIS: Absolutely. I can jump back to the catalog or get my total page. More store procedures are going to execute calculating tax, shipping charges and showing me what my total bill due is. Here is where we're going to add some merchant server capability. Today I would enter credit-card information it's encrypted with SSL. This does not flow in the clear, it's encrypted when it's transmitted. We'll improve this order. I see all prior orders that I've placed with this site.

TODD WARREN: Someone wouldn't need to call a customer receivers representative.

DAVID HARRIS: One of the things that we learned by automating a large number of clients on the Internet is you have to do things differently. You have to provide more feedback to buyers, we also trigger e-mail confirmation of the shipment when that fish leaves the loading dock telling them that the fish is on the way. Of course, we include the air bill 1tracking number for UPS or FedEx.

TODD WARREN: So it calls you to send that message out. Once it sent out you use MAPI to send the confirmation over the Internet.

DAVID HARRIS: That's correct. That's a new category. Web Alert. Web Alert is the beginning of intelligent spamming. We can set out collection notices, notices for things on sale. Now, this order is fed into their existing 2ccounting system which is running on a LAN. We can jump into that accounting system. We see that it's coming in. This is what can be transported into any other existing accounting system that's in place.

TODD WARREN: So you've taken the data in the accounting system and really made work to expand the customer's business. Look like a super application. Thanks a lot.

DAVID HARRIS: Thank you.

TODD WARREN: So, it can be done today. You can take your application that's been designed for the intranet and help your commerce expand to the Internet and offer a great set of new features with what we've done with BackOffice. So I just want to take a second to recap the BackOffice logo, which is a great way for your application once you've taken advantage of these services to tell customers that you're using the BackOffice in an efficient and reliable way. For server applications, we ask that you run as a service, or if you're an ISAPI application being a DOL gives you that part of the requirement. We ask that you be network-dependent in running the system and provide unified logon. We also ask that you be deployable using the management server. If you're a client application, and you have a Windows-based client, using any of the APIs, for example, having sending post on the menu helps you get the logo and making yourself installable with system's management server.

We have a lot of applications logos today. If you're leveraging exchange to get testing for the logo and we expect to more than double the total number of people who have the logo. In summary, Microsoft BackOffice makes it easy for you to leverage the Internet, to create easy-to-manage sites, to take traditional client server application and add additional functionality that gives your customers the presence on the Internet with exchange. There's a lot of functionality to link to Internet mail and we're adding things to make it even easier to access that in a Web-based environment or in a POP 3 and in a TP environment. We designed a BackOffice to make your application or site much more powerful much easier to manage and there's lots of great services to come. Thanks a lot and I look forward to all of you going and getting the BackOffice logo. See you at the breakout.   (Applause.)