39 TCP/SNA

September 2000

update

In this issue

TCP/SNA Update

Published by
Xephon
27-35 London Road
Newbury
Berkshire RG14 1JL
England
Telephone: 01635 38342
From USA: 01144 1635 38342
E-mail: fionah@xephon.com

North American office
Xephon
Post Office Box 350100
Westminster, CO 80035-0100
USA
Telephone: (303) 410-9344

Subscriptions and back-issues
A year's subscription to *TCP/SNA Update*, comprising four quarterly issues, costs £130.00 in the UK; $190.00 in the USA and Canada; £136.00 in Europe; £142.00 in Australasia and Japan; and £140.50 elsewhere. In all cases the price includes postage. Individual issues, starting with the March 1991 issue, are available separately to subscribers for £33.00 ($48.00) each including postage.

Editorial panel
Articles published in TCP/SNA Update are reviewed by our panel of experts. Members include John Bradley (UK), Carlson Colomb (Canada), Anura Gurugé (USA), Jon Pearkins (Canada), and Tod Yampel (USA).

Editor
Fiona Hewitt

Contributions
Articles published in *TCP/SNA Update* are paid for at the rate of £170 ($260) per 1000 words for original material. To find out more about contributing an article, please contact us and we will send you a copy of our *Notes for Contributors*, or you can download a copy from www.xephon.com/contnote.html.

TCP/SNA Update on-line
Code from *TCP/SNA Update* can be downloaded from our Web site at http://www.xephon.com/tcpsnaupdate.html; you will need the user-id from your address label.

Printed in England.

Network performance tuning

I've recently completed some work for a company with a large network of NT users attached to a mainframe. The problem was one of disappointing network performance. The manager was certain that the mainframe network was working fine, and had a plethora of NetView statistics to show how well everything was running. He said that it was the NT users who were complaining, and he maintained right from the beginning that they were also the cause of the problem. (As you might have guessed, he was from the old school of MVS mainframe operators who had, over time, worked his way up into quite a senior management position.)

SNA

The manager was right in so far as access times on all parts of the SNA/VTAM network seemed fine. The right versions of everything were installed, and they were all compatible. The problem didn't appear to be with the mainframe or its users.

OTHER PROTOCOLS

The other protocols in use were:

- TCP/IP (and NetBIOS over TCP/IP)
- IPX/SPX
- NetBEUI/NetBIOS.

TCP/IP was the main protocol in use, and it seemed to be performing reasonably well in most circumstances.

We immediately recommended that the company should use only TCP/IP. If this proved impossible, however, we said that they should at least ensure that only one protocol was run on a server. There are three reasons for this:

- First, each protocol requires processor time to interpret incoming

data packets and send data to the network. Having more than one protocol would therefore slow everything down.

- Second, the LAN card has its own processor, thus reducing the load on the server. It also provides buffering, to improve performance. In a multi-protocol environment, each protocol requires a portion of these resources on the card. Running two or more protocols splits the resources equally between them, which inevitably degrades the amount of time and buffering provided to each of the protocols, and therefore reduces overall response times.
- Third, the buffers in the LAN card reduce the amount of CPU time needed to process incoming packets of data, and also minimize the instances of data lost as a result of the CPU being too busy. Larger buffers will improve performance. If more than one protocol is in use, this will reduce the size of the buffers.

BANDWIDTH MANAGEMENT

We took a look at how the traffic was flowing over the network, and considered the usual options of prioritization schemes, router flow control, and stand-alone bandwidth managers.

Most routers support some form of traffic prioritization. This allows different traffic types to be serviced based on their relative importance – usually based on IP source or destination address and application. Everything seemed OK here.

Adding Quality Of Service (QOS) to TCP/IP is very useful for sites running WAN links, where different traffic types contend for scarce bandwidth. Different vendors implement this in different ways. Again, everything seemed in order.

The company didn't have any bandwidth management products. These usually run on stand-alone edge devices placed between the intranet and the router, and affect outbound and inbound TCP traffic. They change TCP window sizes (the number of bytes transmitted before an acknowledgement is required) and transmit TCP acknowledgements at a controlled rate.

DATA CACHEING

It was with data cacheing that we had our first success. SCSI controllers have a considerable amount of intelligence and memory to speed up the performance of SCSI disk drives. But the default settings often don't provide the best performance. We turned on write cacheing, and, although this didn't directly improve transfer performance, it did reduce the CPU overhead. The CPU was hit heavily by data transfers; our change produced a big improvement in the response times of applications.

The company already had proxy servers. These can reduce inbound intranet traffic by storing frequently-requested resources. And, because data is stored much closer to the end users who request it, response times are also improved.

Hierarchical cacheing is a technique that allows a group of caches to speed performance even more. There are two types of hierarchical cache – sibling and parent-child.

DATA COMPRESSION

Most data flows can be optimized by compressing them. A look at the inefficient packet structures of many TCP/IP applications shows why compression is needed. For example, Telnet transmits only one character (one byte) per 64-byte packet. And many other applications use a large number of small packets, resulting in high overhead per unit of data transferred. Compression generally works best only on lower-speed WAN links (typically 2Mb/sec or less).

We looked at both of the common methods of compression – compressing the TCP header and compressing the entire packet.

However, the router vendor had included compression as part of the software.

THOSE END USERS

By this stage, although we'd made some improvements, we'd basically discovered that most of the optimizing techniques were already in place.

We were beginning to think that the network manager was right – it was all the end users' fault after all. So our next task was to look at the settings on those NT machines and find out exactly how the users were using them.

We looked at parameters affecting TCP/IP stacks. We examined the values specified for the TCP window size and Maximum Transmission Unit (MTU, the largest allowable packet size). It's important not to use large window sizes on slow or error-prone links, where re-transmissions caused by missed acknowledgements will reduce performance. We were also aware that applications that send many small packets (such as HTTP) don't benefit from larger MTUs.

Many users were sending queries to SQL Server, which interprets them first, before retrieving the appropriate data and returning it. One good thing about SQL Server is that it allows users to pre-compile queries, which improves their execution speed. However, from our point of view, the size of the data sent back to a workstation can be greatly reduced by executing most of the processing on the server and just returning the result data.

We found that users were carrying out a lot of printing. Print files can impose a huge burden on network bandwidth. This is because documents can contain large amounts of image data, graphics, and a variety of embedded fonts.

We therefore suggested that users should be encouraged to reduce the size of their print files. We asked them to consider using vector rather than bitmapped graphics. These are much smaller, because they contain a description of what to print rather than a series of dots that need to be printed. Most printers can convert vector files into bitmaps, considerably reducing the amount of data that needs to be sent. Using Postscript and HPGL can considerably reduce network loading, and we recommended that they use a printer that supports them.

We also looked at the way that back-ups were carried out. Back-ups place a big load on the server and impact the response times experienced by users, particularly if they're carried out at peak times. The frequency of back-ups depends on the value of the data and the resilience of the storage. Where RAID storage is used with the highest level of protection, back-ups are probably required less frequently.

It's possible to back up locally on the server or remotely to another system equipped with a back-up device. Backing up remotely is slower than backing up locally, and still places a heavy I/O burden on the server. A local back-up runs more quickly and reduces the time during which the server is stressed. However, with a remote back-up the network is stressed. We found that the way back-ups were being performed was impacting on the performance experienced by users.

CONCLUSION

In many ways, the network manager was right – it *was* the users who were causing the problem. They weren't using SQL Server as efficiently as they could, they were printing too many large files with the 'wrong' type of embedded graphics, and they were carrying out too many back-ups at the 'wrong' time of day. The network was performing almost as well as it could, and the servers were generally configured correctly.

In conclusion, then, when you plan to carry out a major project on performance, it's worth bearing in mind that conducting a little end-user education instead could be quicker, and could result in long-term performance improvements.

Nick Nourse
Independent Consultant (UK)

Code from *TCP/SNA Update* articles

As a free service to subscribers and to remove the need to rekey the scripts, code from individual articles of *TCP/SNA Update* can be accessed on our Web site, at

http://www.xephon.com/tcpsnaupdate.html

You will need the user-id shown on your address label.

SNA APPC general tuning guidelines

This article presents a number of guidelines for tuning your SNA APPC implementation. Note that the recommendations given will be helpful whatever the platform in use, but that the actual tuning may vary.

LU VERSUS APPC

The first key point to highlight is that you should use Logical Unit 6.2 (LU 6.2), also known as APPC, where possible. This is particularly relevant for bulk data transfer applications that don't use the APPC API. You should either migrate these to APPC or replace them with applications that can use APPC. All new applications, whether in-house or packaged, should be designed or purchased to use APPC. There are many differences between LU 6.2 and older Logical Unit types such as LU 2 and LU 0. These are outdated technologies; if you have to use SNA, then LU 6.2 is what you need.

Despite this, LU2 is still prominent in many standard IBM-type installations. But it is misused as a data transfer protocol – even in the 'old' days, you could get more efficient transfer with LU 0. LU 2 is ideally suited to terminal traffic. LU2 devices could (and can) send only 32KB of data at any one time, and must simulate an ENTER key operation to simulate every record the client has to send. This, combined with the high overhead of control data required to implement the data stream, means that LU 2 is limited and inefficient when moving large amounts of data.

APPC, on the other hand, has features that allow large send and receive record sizes – the architecture allows up to 2GB of data to be sent or received at one time. Some products allow up to 64KB to be transferred and offer enhanced data buffering, enabling data to be moved at high rates. In addition, the control overhead for the protocol is low once a conversation has been established.

IBM provides a number of documents to facilitate migration from LU 2 to LU 6.2, including:

- *APPC Design for HLLAPI and 3270 Programmers*
- *HLLAPI or APPC: Design Criteria*
- *CICS 3270 to APPC Client/Server: A Migration Methodology*

USING MODES FOR DATA TRANSFER APPLICATIONS

The next key area to discuss is the use of modes for data transfer applications. IBM's mode tables may seem a simple subject, but, used wisely, a mode with a large Request/Response Unit size and large pacing window values with a class of service of 'medium' or 'low' will make a difference to the way data is transferred.

IBM provides a default mode of #BATCH as an example, and this is an ideal starting point. SNA is excellent at providing options and customization values for individual data sessions, and this area can be heavily tuned. Characteristics such as Request/Response Unit size, pacing window values, transmission priority, and security requirements are all held in modes and are part of the systems mode tables. Tuning this area can be complex for the uninitiated but is well worth the effort, as it stops bandwidth-hungry applications from dominating the network. (It's rather amusing that SNA provides capabilities that router/IP networks can still only dream about.) It is vital that bulk data transfer applications use large Request/Response Unit sizes, large pacing windows, and have low transmission priorities – change these characteristics to anything other than what's mentioned here, and your on-line systems will suffer.

Calculating the Request/Response Unit size

You need to ensure the Request/Response Unit size is correct. It has to be a multiple of the Data Link Control frame size. This increases link throughput. It is essential that an entire Request/Response Unit can fit into one Data Link Control frame, otherwise segmentation will occur. You must subtract the Data Link Control frame header from the overall frame size to determine the appropriate Request/Response Unit size. Some applications assist in this area, but not all. If you do make the Request/Response unit size larger than the Data Link

Control frame size, then select a size that is as close as possible to an even multiple of the Data Link Control frame size. There are normally up to nine bytes of header information in a Request/Response Unit for APPN networks. For subarea data traffic, this increases to twenty-nine bytes. The recommended calculations are shown below:

- For subarea networks:

  RU size = an even multiple of DLC frame size minus 29.

- For APPN networks:

  RU size = an even multiple of the DLC frame size minus 9.

There is one exception to these rules. This is on AS/400 systems, where the system calculates the most efficient Request/Response Unit size for the user in question.

SNA products are generally distributed with default Request/ Response Unit sizes of less than 2KB. This is ideal for interactive traffic and users with small data transfer requirements. For bulk data transfers, however, it is no good at all. 4KB is preferable, and if you increase single session throughput to as high as 64KB, you will improve performance.

Note that we're only talking theory here, and that, as always, nothing is free and there's a trade-off to be considered. Increasing the size of Request/Response Units for a machine increases the amount of memory needed to store and forward those Request/Response Units. This is extremely detrimental to the system that is the data receiver, which must have enough memory to hold the whole window of Request/Response Units that it has to handle. Data senders, on the other hand, require only one Request/Response Unit buffer.

Segmentation of Request/Response Units will also occur if the Request/Response Unit cannot be stored into a single Data Link Control frame for transmission. The Request/Response Unit is broken into segments small enough to fit into the frame and the receiver has to re-assemble them. Although segmenting should be used wherever possible, bear in mind that any other tuning operation should be

completed before segmentation takes place. It's best to increase the Data Link frame size and pacing windows first.

PACING

The next area of tuning I'd like to discuss is the one that many people find the most confusing: pacing.

If an SNA product supports adaptive pacing, you should always use the maximum allowable Request/Response Unit size. You also need to ensure the window size is increasing to at least eight; if it isn't, you have resource constraints or a configuration issue somewhere along the data paths.

If the product you're tuning supports only fixed pacing, you should use a window size of eight to start off with, and then increase it to see if performance improves. If it doesn't, go back to eight as a window size. If your pacing size is inadequate, then for APPC sessions both the server and client machines will wait for data acknowledgements to travel across the network. Larger pacing window sizes will help eliminate send and wait behaviour when transferring data. If data can be sent continually while acknowledgements are received, network capacity can be used efficiently.

SNA uses both fixed and adaptive pacing algorithms. Adaptive pacing is the newer and more advanced technique, and products and services that support adaptive pacing can normally use fixed pacing if the receiver or sender they're working with doesn't support adaptive pacing. The decision on the type of pacing used with sessions is automatically negotiated between two machines when the session is being established. It's not a user-configurable option but part of the systems programmer's definition and set-up.

If two communicating machines are using fixed pacing, a large pacing value is usually better than a smaller one. You need to plan on memory consumption when setting up pacing, as pacing values affect buffers that are required for session windows. The window size stays constant if fixed pacing is being used.

Remember that pacing helps performance, but only to a point – it can have a detrimental effect if it's increased too far. You need to experiment to find the ideal for your network. And, since all networks vary, and data loads can vary by time of week or month, you have to try to get the optimum for all possible configurations. Ideally, you shouldn't set aside too many buffers, because if they aren't used you waste memory. And remember that in today's SNA environments, with dynamic reconfiguration, it's possible to have one set of parameters for daytime workloads and another for evening and night shifts, by using automation to load different pacing and buffer values at certain times.

Adaptive pacing is intuitive, and uses memory size and availability to automatically determine the best window size for sessions at a specific point in time. The receiving machine's memory size will always limit the pacing window's size. The adaptive pacing algorithms are very efficient as they use only what memory is available at the time of the transfer; however, this can mean unpredictable transfer times if other systems are using a lot of memory at one time and not at another. Adaptive pacing can be harder to tune in some cases. Adaptive pacing may be efficient but many other systems are not. Machine storage and tuning is important if you use adaptive pacing. It's also essential that physical memory and available buffer constraints are taken into account on all machines in the data transfer path – these affect the size of the pacing window for the session in question.

Both fixed and adaptive pacing allow for different pacing window size values, based on the direction of flow. If a machine can send eight frames before requiring some form of acknowledgement, it's possible it can receive only one between acknowledgements. This is a key point, as you need to ensure before increasing or decreasing pacing window sizes that both the send and receive partners are reviewed and the values adjusted accordingly, otherwise responses times can become erratic. Consistent response times should always be your goal.

Take, for example, a machine (1) that uses fixed pacing with a send window of eight and a receive Request/Response Unit size of sixty-three bytes. When it sends data, the sending pacing window will

govern the number of Request/Response Units it can send before it gets an acknowledgement saying it's ready to send more data. The machine can send only eight frames before receiving an acknowledgement. If the acknowledgement is not received before the eighth frame is sent, the machine has to stop and idle until it's received. When the machine is receiving data, the receive pacing window will govern how many Request/Response Units the machine can receive. The sending machine will send only sixty-three Request/Response Units, unless machine (1) states that it wants to receive more frames. It will send an acknowledgement for every sixty-third Request/Response Unit sent.

MODES

One problem that most network systems programmers have encountered is that they think an application is using one set of mode values when it's actually using a completely different set of values. It's important to determine and be aware of what modes are being used for data transfer applications. You also need to check their consistency as data is transferred across multiple networks.

SNA profiles use modes to determine a session's characteristics. However, a #BATCH mode in one network may be different in another network that your data has to cross. It's important, particularly in multiple networks, that systems programmers ensure consistency, but this involves communicating what they're doing to other people, which isn't always a great systems programmers' trait.

There are, however, some easy ways to ensure consistency. The first is to set up applications so that they use one of the five supplied user modes on their allocate commands. These modes are:

- BLANK
- #BATCH
- #BATCHSC
- #INTER
- #INTERSC.

Since these modes ship with many SNA applications, you build in consistency by using them. Talk to each site where modes can be defined, and agree on a pre-defined set-up. Once consistency is achieved, ensure that all definitions are propagated across the network. This means defining the mode in all products that do, and might, use it. Inconsistency can cost. When a mix of logical units share the same modes, it's not always clear where the governing mode is coming from. In some cases, it comes from the definition of the bind for the initiating logical unit. In others, it can be taken from the definition for the receiver of a bind. This means binds from 1 to 2 and from 2 to 1 could be different if no consistency is in place.

PRODUCT-SPECIFIC TUNING RECOMMENDATIONS

Below, we run through some tuning recommendations that are product-specific.

Assembler macroinstruction statements control the Network Control Program (NCP). These are commonly referred to as NCP definition statements. The following are some key factors that affect performance:

- The Request/Response Unit size is affected by the MAXBFRU and UNITSZ parameters. For each NCP in your network, you should:

  1 Set the UNITSZ to 4000 bytes.

  2 Determine the size of the largest Path Information Unit (PIU) in your network. This will be a little smaller than the size of the Data Link Control frames the NCP receives from any other machines.

  3 Divide the Path Information Unit size from (2) by the UNITSZ value in (1) and then round up to the nearest integer. This determines what you must code for the MAXBFRU value.

  4 Check to make sure that MAXDATA in VTAM and NCP is greater than the UNITSZ multiplied by the MAXBFRU, and

that this in turn is greater than the Path Information Unit Size.

(MAXDATA > UNITSZ*MAXBFRU > PIU size)

- The number of buffers that are reserved by the NCP for receiving a Request/Response Unit is specified by MAXBFRU. Each buffer is a UNITSZ in length. When an NCP receives data, it will fill these buffers up waiting for the data as it arrives. If you have too many buffers per Request/Response Unit, unused buffers will be wasted and the NCP will wait unnecessarily. The NCP actually waits until all buffers specified in the MAXBFRU parameter are filled before any data is forwarded on to another NCP or VTAM. If the buffers don't fill, data is transferred when a particular time value expires. However, this waiting is very inefficient and should not occur at any cost.
- Remember as well that the size of the buffers affects performance. Larger buffers allow the transfer of bulk data to be more efficient than smaller buffer sizes. The BFRS NCP definition statement is used to define the NCP's main memory buffer size. This should always be set to a value equal to 240.
- The SLODOWN statement is used to define a threshold that tells the NCP to enter a protective mode of operation known as 'slow down'. This occurs if the NCP thinks it will run out of storage. The parameter specifies a percentage value, so that if you code ten as a value, you're telling the NCP to enter slow down state when the memory available is only ten percent free for new data. IBM recommends a value of six for this parameter.
- The Delay parameter tells VTAM how long data should be held before it's transmitted to the NCP. This timer affects how pacing acknowledgements and confirmation messages get back to the NCP-attached systems. If it's set too high, performance will be degraded. Ideally, it should be set to zero.
- A specialist area of the NCP is the Token Ring adapter. The NCP can have two types of Token Ring adapter, each with different capabilities:

- A TIC type 1 uses 2576 bytes of storage and operates at a speed of only four megabits per second.
- A TIC type 2 has 64 kilobytes of storage and can operate at speeds of 16 or 4 megabits per second.

The data link control frame size is controlled by a number of parameters:

- The RCVBUFC parameter controls the inbound Data Link Control frames to the NCP. This parameter must be set large enough to receive two whole Data Link Control frames from any peripheral attached node. If it's set too low, retransmissions normally occur. If an end station sends a frame that's too large, disconnections occur. It should be set twice as large as the MAXTSL value. The maximum value for a TIC type 1 adapter is 4095 bytes, and 32,000 bytes for a TIC type 2 adapter.
- The MAXTSL parameter affects the size of the Token Ring Data Link Control frame that the NCP can send to and receive from other Token Ring stations/peripherals. IBM issues the following guidelines:
  - Never exceed a value of 1108 bytes for a TIC type 1 adapter.
  - Use 4060 bytes for TIC type 2 adapters running at 4 megabits per second.
  - Use 16,000 bytes for TIC type 2 adapters running at 16 megabits per second.
- Data Link Control pacing is affected by the T2TIMER definition statement, which is used to specify three different values for use by the NCP:
  - T1, which is the acknowledgement time-out value for local LAN connections.
  - T2, which is the acknowledgement time-out for remote connections. These are connections to partner devices on a different physical segment of the local area network.

- W, which is the receive acknowledgement count T2TIMER=(T1,T2,W). When the NCP receives Data Link Control frames, it keeps a timer and count active for every link. Either the timers will expire or the number of frames will equal the value coded in W. In both cases, a link level acknowledgement is then sent for all the frames that have been received since the NCP last sent an acknowledgement. W is actually what's termed the receive pacing window size. The NCP treats the W parameter differently depending on the connection type. For subarea type connections, the value is used as the window acknowledgement size. For peripheral connections, the NCP uses only two window sizes, 1 and 2. If the T2TIMER statement is specified for a peripheral connection, a value of two is used. If not specified at all in the NCP definition, a value of one is used.

The following are guidelines:

- Use the T2TIMER operand on the LINE and BUILD NCP definition statements for peripheral connections when they have machines that support a SEND window of greater than one. For example, code T2TIMER=(1,1,2).
- If the NCP is connected to a lot of peripheral stations that do not support a SEND window of greater than one, use a local time-out of less than 0.002 (200 milliseconds), or don't code T2TIMER. This stops the end stations from waiting too long for the NCP to send an acknowledgement.
- For subarea nodes, use larger values for the T2TIMER operands. You'll need to monitor in order to discover what these values should be (and you'll probably find you need to devote your life to getting an optimum set-up).

– The MAXOUT definition statement specifies the number of Data Link Control frames that the NCP can send before a

Data Link Control acknowledgement must be returned. The value for this parameter can be determined only by the capacity of the receiving machine. It should be as large as is possible on the type of link being used.

- VTAM tuning is performed by making changes to a number of files on the mainframe computer executing VTAM. The ideal reference for such changes and finding out what the definition statements mean is the *VTAM Resource Definition Guide*.
- Request/Response Unit sizes are coded as a base and mantissa on the RUSIZES statement in the VTAM MODEENT macro-instruction. These instructions make up part of the VTAM mode table, a sample of which is shown below. The value RUSIZES=X'8787' in the first entry says that both the secondary logical unit to primary logical unit and the primary logical unit to secondary logical unit would be eight multiplied by two to the power seven ($8*2^7$), or 1,024 bytes.

  For bulk data transfer operations, code RUSIZES=X'8D8D'. This is a sixty-four byte RUSIZE.
- The IOBUF operand tells VTAM how big to make the buffers it uses for sending data for channel-attached devices. The larger the buffer size, the less processing required to send and receive large amounts of data to and from channel-attached devices. Ideally, you should set this to a value of 4000 bytes.

SAMPLE VTAM MODE TABLE

```
TITLE     'SMDNDMØØ STD MODE TABLE NDM.'
*
*        ************************************************************
*        *                                                          *
*        *                                                          *
*        ************************************************************
*
SMDNDMØØ MODETAB
*
* ENTRY USED WITH OTHER NDM-MVS NODES
*
```

```
NDMLOGM  MODEENT LOGMODE=NDMLOGM,  ENTRY NAME                          X
               TYPE=1,             NON-NEGOTIABLE BIND                 X
               FMPROF=X'Ø4',       FUNCTION MGMT. PROFILE 4            X
               TSPROF=X'Ø4',       TRANS SERVICE  PROFILE 4            X
               PRIPROT=X'B3',      PRIMARY PROTOCOL                    X
               SECPROT=X'B3',      SECONDARY PROTOCOL                  X
               COMPROT=X'6Ø8Ø',    COMMON PROTOCOL                     X
               PSNDPAC=X'Ø6',      PRIMARY SEND PACING                 X
               SRCVPAC=X'Ø6',      SECONDARY RECEIVE PACING            X
               SSNDPAC=X'Ø6',      SECONDARY SEND PACING               X
               RUSIZES=X'8787',    4K MAX RU SIZE FOR PRI AND SEC      X
               COS=LOW,                                                X
               PSERVIC=X'ØØØØØØØØØØØØØØØØØØØØØØØØ'
*
* ENTRY USED WITH LUØ.
*
NDMPCLUØ MODEENT LOGMODE=NDMPCLUØ, ENTRY NAME                          X
               FMPROF=X'Ø4',       FUNCTION MGMT. PROFILE 4            X
               TSPROF=X'Ø4',       TRANS SERVICE  PROFILE 4            X
               PRIPROT=X'B1',      PRIMARY PROTOCOL                    X
               SECPROT=X'B1',      SECONDARY PROTOCOL                  X
               COMPROT=X'7Ø8Ø',    COMMON PROTOCOL                     X
               PSNDPAC=X'Ø7',      PRIMARY SEND PACING                 X
               SRCVPAC=X'Ø7',      SECONDARY RECEIVE PACING            X
               SSNDPAC=X'Ø7',      SECONDARY SEND PACING               X
               RUSIZES=X'8686',    4K MAX RU SIZE FOR PRI AND SEC      X
               COS=LOW,                                                X
               PSERVIC=X'ØØØØØØØØØØØØØØØØØØØØØØØØ'O
*
* ENTRY USED WITH LUØ.
*
M32782S  MODEENT LOGMODE=M32782S,  ENTRY NAME                          X
               FMPROF=X'Ø4',       FUNCTION MGMT. PROFILE 4            X
               TSPROF=X'Ø4',       TRANS SERVICE  PROFILE 4            X
               PRIPROT=X'B1',      PRIMARY PROTOCOL                    X
               SECPROT=X'B1',      SECONDARY PROTOCOL                  X
               COMPROT=X'7Ø8Ø',    COMMON PROTOCOL                     X
               PSNDPAC=X'Ø7',      PRIMARY SEND PACING                 X
               SRCVPAC=X'Ø7',      SECONDARY RECEIVE PACING            X
               SSNDPAC=X'Ø7',      SECONDARY SEND PACING               X
               RUSIZES=X'8686',    4K MAX RU SIZE FOR PRI AND SEC      X
               COS=LOW,                                                X
               PSERVIC=X'ØØØØØØØØØØØØØØØØØØØØØØØØ'
*
* ENTRY USED WITH NDM-PC/IRMA
*
D4A32782 MODEENT LOGMODE=D4A32782, ENTRY NAME                          X
               FMPROF=X'Ø3',       FUNCTION MGMT. PROFILE 3            X
               TSPROF=X'Ø3',       TRANS SERVICE  PROFILE 3            X
```

```
               PRIPROT=X'B1',      PRIMARY PROTOCOL                    X
               SECPROT=X'90',      SECONDARY PROTOCOL                  X
               COMPROT=X'3080',    COMMON PROTOCOL                     X
               RUSIZES=X'87C7',    1K SEND RU, 1.5K RECEIVE RU         X
               COS=LOW,                                                X
               PSERVIC=X'02000000000001850000007E00'
*
* ENTRY USED WITH NDM-PC WITH STRUCTURED FIELD SUPPORT
*
NDMPCSF  MODEENT LOGMODE=NDMPCSF,  ENTRY NAME                          X
               FMPROF=X'03',       FUNCTION MGMT. PROFILE 3            X
               TSPROF=X'03',       TRANS SERVICE  PROFILE 3            X
               PRIPROT=X'B1',      PRIMARY PROTOCOL                    X
               SECPROT=X'90',      SECONDARY PROTOCOL                  X
               COMPROT=X'3080',    COMMON PROTOCOL                     X
               RUSIZES=X'87F8',    1K SEND RU, 3840 RECEIVE RU         X
               COS=LOW,                                                X
               PSERVIC=X'02800000000001850000007E00'
*
* ENTRY USED WITH NDM-PC COAX ATTACHED TO A NON-SNA LOCAL CONTROLLER
*
NDMPCLC  MODEENT LOGMODE=NDMPCLC,  ENTRY NAME                          X
               FMPROF=X'02',       FUNCTION MGMT. PROFILE 3            X
               TSPROF=X'02',       TRANS SERVICE  PROFILE 3            X
               PRIPROT=X'71',      PRIMARY PROTOCOL                    X
               SECPROT=X'40',      SECONDARY PROTOCOL                  X
               COMPROT=X'2000',    COMMON PROTOCOL                     X
               RUSIZES=X'87C7',    1K SEND RU, 1.5K RECEIVE RU         X
               COS=LOW,                                                X
               PSERVIC=X'02000000000001850000007E00'
*
* ENTRY USED WITH NDM-VMS AND NDM-TANDEM
*
NDMVMS   MODEENT LOGMODE=NDMVMS,                                       X
               TYPE=1,                                                 X
               FMPROF=X'04',                                           X
               TSPROF=X'04',                                           X
               PRIPROT=X'B1',                                          X
               SECPROT=X'B1',                                          X
               COMPROT=X'7080',                                        X
               SRCVPAC=X'04',      SRCVPAC,SSNDPAC, AND PSNDPAC        X
               SSNDPAC=X'04',      VALUES CAN BE TUNED                 X
               PSNDPAC=X'04',                                          X
               RUSIZES=X'8888',    2K SEND AND RECV RUSIZE             X
               COS=LOW,                                                X
               PSERVIC=X'000000000000000000000000'
*
* ENTRY USED WITH NDM-400 SNUF (LU 0)
```

```
*
SNUF2K   MODEENT LOGMODE=SNUF2K,                                       X
               FMPROF=X'Ø4',                                           X
               TSPROF=X'Ø4',                                           X
               PRIPROT=X'B1',                                          X
               SECPROT=X'B1',                                          X
               COMPROT=X'7Ø8Ø',                                        X
               SRCVPAC=X'ØØ',      SRCVPAC,SSNDPAC, AND PSNDPAC        X
               SSNDPAC=X'ØØ',      VALUES CAN BE TUNED                 X
               PSNDPAC=X'ØØ',                                          X
               RUSIZES=X'8888',    2K SEND AND RECV RUSIZE             X
               COS=LOW,                                                X
               PSERVIC=X'000000000000000000000000'
*
* LU 6.2 LOGMODE
*
NDM624K  MODEENT LOGMODE=NDM624K,                                      X
               TYPE=1,                                                 X
               FMPROF=X'13',                                           X
               TSPROF=X'Ø7',                                           X
               PRIPROT=X'BØ',                                          X
               SECPROT=X'BØ',                                          X
               COMPROT=X'DØB1',                                        X
               RUSIZES=X'8989',    4K SEND AND RECV RUSIZE             X
               COS=LOW,                                                X
               PSERVIC=X'060200000000000000000300'
*
NDM621K  MODEENT LOGMODE=NDM621K,                                      X
               TYPE=1,                                                 X
               FMPROF=X'13',                                           X
               TSPROF=X'Ø7',                                           X
               PRIPROT=X'BØ',                                          X
               SECPROT=X'BØ',                                          X
               COMPROT=X'5ØB1',                                        X
               RUSIZES=X'8D8D',   64K BYTES RU SIZE                    X
               COS=LOW,                                                X
               PSERVIC=X'060200000000000000000000'
*
NDM622K  MODEENT LOGMODE=NDM622K,                                      X
               TYPE=1,                                                 X
               FMPROF=X'13',                                           X
               TSPROF=X'Ø7',                                           X
               PRIPROT=X'BØ',                                          X
               SECPROT=X'BØ',                                          X
               COMPROT=X'5ØB1',                                        X
               RUSIZES=X'8888',    2K SEND AND RECV RUSIZE             X
               COS=LOW,                                                X
               PSERVIC=X'060200000000000000000000'
```

```
*
* SNA SERVICES MANAGER MODE FOR LU 6.2
*
* (THIS LOGMODE IS FOR VTAM USE, THE USER MUST DEFINE THIS IN THE
*  MODE TABLE BUT DO NOT SPECIFY THIS AS THE LOGMODE IN THE NDM
*  NETMAP)
*
SNASVCMG MODEENT LOGMODE=SNASVCMG,                                    X
               TYPE=1,                                                X
               FMPROF=X'13',                                          X
               TSPROF=X'Ø7',                                          X
               PRIPROT=X'BØ',                                         X
               SECPROT=X'BØ',                                         X
               COMPROT=X'DØB1',                                       X
               RUSIZES=X'8585',    4K SEND AND RECV RUSIZE            X
               PSERVIC=X'060200000000000000000300'
*
         MODEEND
*
         END
```

FINAL WORD

Although tuning any type of network is a time-consuming and never-ending task, many performance benefits can be gained by putting in a little time and effort. In this article, I've suggested areas that you should explore when tuning your network, while also demonstrating that the enormity of the task should not be underestimated.

Elizabeth Bradley
Network Systems Programmer (UK)

NvDM send procedure

Here, we present the sample tables, skeletons, and programs, and the ISPF panels for the NvDM send procedure published in the last issue of *TCP/SNA Update* (June 2000).

NVDM SEND PROCEDURE: SAMPLE TABLE

```
*  NvDM Send Procedure
*  Table NSPTØØØ to management project and relative destinations.
*
*                                    NvDM Node
* Alias Projects              TR MI RM PA FI GE NA1 NA2
* ------------------------------------------------------
  NDMADM                      S  S  S  S  S  S  S   S
  TEMP                        _  _  _  _  _  _  _   S
  PROJa                       _  S  _  _  _  _  _   S
  PROJb                       S  _  S  _  _  S  S   S
  PROJc                       _  _  _  S  S  _  S   S
  XXXXXXXX                    _  _  _  _  _  _  _   _
```

NVDM SEND PROCEDURE: SAMPLE TABLE LIST OF NVDM DESTINATIONS

```
*  NvDM Send Procedure
 *  Table NSPTØ1Ø to NvDM nodes Destinations
*
   =1=    TRENTO     NDMTRE1
   =2=    MILANO     NDMMIL1
   =3=    ROMA       NDMROM1
   =4=    PALERMO    NDMPAL1
   =5=    FIRENZE    NDMFIR1
   =6=    GENOVA     NDMGEN1
   =7=    NAPOLI1    NDMNA11
   =8=    NAPOLI2    NDMNA21
    *     LOCALNDM   NDMNAØØ
```

NVDM SEND PROCEDURE: SAMPLE SKELETON TO GENERATE JOBS OF NVDM UTILITIES

```
//         JOB (ZZNSØØØØ),NOTIFY=NDMOPR1,CLASS=S,MSGCLASS=X,
//     MSGLEVEL=(1,1)
//JOBLIB   DD  DSN=NDM.V1R6M2.SFZDLOAD,DISP=SHR
//STEP1   EXEC PGM=DSXPREP,REGION=3ØØØK,
```

```
//        PARM='FUNCTION=*FUNCT*,USERID=NDMOPR1,PASSWORD=NDMPWD'
//DSXDRD   DD DSN=NVDM162.TCPE.NDMDRD,DISP=SHR
//DSXGIX   DD DSN=NVDM162.TCPE.NDMGIX,DISP=SHR
//DSXGIXD  DD DSN=NVDM162.TCPE.NDMGIXD,DISP=SHR
//NDMRQFDA DD DSN=NVDM162.TCPE.NDMRQFDA,DISP=SHR
//NDMRQF   DD DSN=NVDM162.TCPE.NDMRQF,DISP=SHR
//DSXTCF   DD DSN=NVDM162.TCPE.NDMTCF,DISP=SHR
//DSXLIB   DD DSN=NVDM162.TCPE.NDMLIB,DISP=SHR
//DSXLIBT  DD DSN=NVDM162.TCPE.NDMLIBT,DISP=SHR
//DSXHFDI  DD DSN=NVDM162.TCPE.NDMHFDI,DISP=SHR
//DSXHFDA  DD DSN=NVDM162.TCPE.NDMHFDA,DISP=SHR
//SNAP     DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSPRINT DD DSN=*OUT*,
//  SPACE=(CYL,(1,1,Ø)),DCB=(RECFM=FB,BLKSIZE=133Ø,LRECL=133),
//  UNIT=339Ø,VOL=SER=WRKHØØ,DISP=(NEW,CATLG)
//DSXPRINT DD SYSOUT=*
//SYSIN    DD *
```

NVDM SEND PROCEDURE: SAMPLE PROGRAM TO GENERATE NVDM NUMBER PLAN-ID

```
IDENTIFICATION DIVISION.
       PROGRAM-ID.                      NDMCOUNT.
      *=================================================================*
      *  Update of the NvDM plan Number                                 *
      *=================================================================*
       ENVIRONMENT DIVISION.
       CONFIGURATION SECTION.
       SPECIAL-NAMES.
           DECIMAL-POINT IS COMMA.
       INPUT-OUTPUT SECTION.
       FILE-CONTROL.
      *================= INPUT/OUTPUT FILE & RECORD  ==================*
           SELECT  ZFILE1 ASSIGN TO ZFILE1
           FILE    STATUS      ST-CHECK.
           SELECT ZFILE1O    ASSIGN TO ZFILE1O
           FILE     STATUS     ST-CHECK1.
       EJECT
       DATA DIVISION.
       FILE SECTION.
       FD  ZFILE1 IS EXTERNAL
           RECORDING MODE IS F.
       Ø1 REC-ZFILE1.
          Ø2 TR                       PIC X.
          Ø2 PLANNUM                  PIC 9(3) COMP-3.
          Ø2 FIL                      PIC X(77).
       FD  ZFILE1O
           RECORDING MODE IS F
```

```
           LABEL RECORDS IS STANDARD
                 RECORD 80.
        01 REC-ZFILE1O.
           02 TRO                      PIC X.
           02 PLANNUMO                 PIC 9(3) COMP-3.
           02 FILO                     PIC X(77).
       *================== WORKING STORAGE AREA ========================*
        WORKING-STORAGE SECTION.
        01  ST-CHECK                     PIC X(002)   VALUE SPACES.
            88  END-FILE                              VALUE '10'.
            88  OK-ALL                                VALUE '00'.
        01  ST-CHECK1                    PIC X(002)   VALUE '00'.
        01  TIPREC                       PIC X        VALUE 'N'.
        01 WS-PNUM                      PIC 9(003)    VALUE  999.
        01 WS-PNUM-RES                  PIC 9(003)    VALUE  0.
        01 WS-RET-PNUM                  PIC 9(003)    VALUE  0.
        01 WS-RET-CODE                  PIC 9(007)    VALUE  0.
        01 WS-CONT-HEADER               PIC 9(004)    VALUE  0.
        01 SW-FILE-OPEN                 PIC X(002)    VALUE 'NO'.
           88 FILE-OPEN                              VALUE 'SI'.
       *=================== Fields for STATISTICS and ERRORS ===========*
        01 WS-CAMPI-SEGNALAZ.
           05 WS-RIGA-ASTER             PIC X(112)     VALUE SPACES.
           05 WS-RIGA-ERR-1.
              07 FILLER                 PIC X(005)     VALUE '*    '.
           05 WS-RIGA-ERR-2.
              07 FILLER                 PIC X(050)     VALUE
                 '* Errors condition into program : '.
              07 WS-PGM-ID              PIC X(008)     VALUE SPACES.
              07 FILLER                 PIC X(014)     VALUE ' - From
LABEL '.
              07 WS-LABERR              PIC X(030)     VALUE SPACES.
       *=================== Fields of work =============================*
        01 WS-CTR-READ                  PIC 9(009)     VALUE ZEROS.
        01 WS-CTR-HANDLE                PIC 9(009)     VALUE ZEROS.
        01 WS-CTR-WRITE-A               PIC 9(009)     VALUE ZEROS.
        01 WS-CTR-WRISI                 PIC 9(009)     VALUE ZEROS.
        01 WS-CTR-WRINO                 PIC 9(009)     VALUE ZEROS.
          PROCEDURE DIVISION.
            DISPLAY '*********************************************'.
            DISPLAY '**              INIT EXECUTION             **'.
            DISPLAY '*********************************************'.
       *============= Routines of INIT/RUN/CLOSE ======================*
            PERFORM I0001-INIT-JOB THRU I0001-99.
            PERFORM R0001-READ-FIL THRU R0001-99
                          UNTIL   END-FILE.
            PERFORM C0001-END-JOB  THRU C0001-99.
        I0001-INIT-JOB.
            MOVE 'I0001-INIT-JOB-1'          TO WS-LABERR.
            MOVE 'NDMCOUNT'                  TO WS-PGM-ID.
```

```
*================ Open Files Input/Output  ======================*
     OPEN INPUT ZFILE1.
     IF ST-CHECK NOT EQUAL ZEROS
        DISPLAY ST-CHECK
        DISPLAY WS-RIGA-ASTER
        DISPLAY WS-RIGA-ERR-1
        DISPLAY WS-RIGA-ERR-2
        DISPLAY '*  ERROR IN OPEN FILE ZFILE1'
        DISPLAY '*  STATUS = ' ST-CHECK
        PERFORM  ERRCHK THRU ERRCHK-EX.
     OPEN OUTPUT ZFILE1O.
     IF ST-CHECK1 NOT EQUAL ZEROS
        DISPLAY WS-RIGA-ASTER
        DISPLAY WS-RIGA-ERR-1
        DISPLAY WS-RIGA-ERR-2
        DISPLAY '*  ERROR IN OPEN FILE ZFILE1O'
        DISPLAY '*  STATUS = ' ST-CHECK1
        PERFORM  ERRCHK THRU ERRCHK-EX.
     MOVE 'SI'  TO SW-FILE-OPEN.
  I0001-99.
     EXIT.
 R0001-READ-FIL.
*=================== Read Input File  ============================*
     MOVE 'R0001-READ-FIL'     TO  WS-LABERR.
     READ ZFILE1.
     IF NOT OK-ALL  AND NOT END-FILE
        DISPLAY WS-RIGA-ASTER
        DISPLAY WS-RIGA-ERR-1
        DISPLAY WS-RIGA-ERR-2
        DISPLAY '*  ERROR IN READ  FILE ZFILE1'
        DISPLAY '*  STATUS = ' ST-CHECK
        PERFORM  ERRCHK  THRU ERRCHK-EX.
     IF NOT END-FILE AND PLANNUM EQUAL WS-PNUM
        MOVE WS-PNUM-RES TO PLANNUM.
     IF NOT END-FILE AND PLANNUM NOT EQUAL WS-PNUM
        ADD 1 TO PLANNUM.
     IF NOT END-FILE AND TR EQUAL TIPREC
        MOVE TR TO TRO
        MOVE PLANNUM TO PLANNUMO
        WRITE   REC-ZFILE1O
        ADD 1 TO WS-CTR-WRISI.
     IF NOT END-FILE AND TR NOT EQUAL TIPREC
        ADD 1 TO WS-CTR-WRINO.
     IF END-FILE AND WS-CTR-READ = 0
        DISPLAY '************************************'
        DISPLAY '*         FILE ZFILE1   EMPTY       *'
        DISPLAY '************************************'
        PERFORM  ERRCHK  THRU ERRCHK-EX.
     IF NOT END-FILE
        ADD 1 TO WS-CTR-READ
     IF ST-CHECK1 NOT EQUAL ZEROS
```

```
           ADD  1  TO  WS-CTR-WRITE-A
           DISPLAY WS-RIGA-ASTER
           DISPLAY WS-RIGA-ERR-1
           DISPLAY WS-RIGA-ERR-2
           DISPLAY '*  ERROR IN WRITE FILE ZFILE1O'
           DISPLAY '*  STATUS = ' ST-CHECK1
           PERFORM  ERRCHK  THRU ERRCHK-EX
        ELSE
           IF NOT END-FILE
              ADD  1  TO  WS-CTR-HANDLE.
     R0001-99.
        EXIT.
    ERRCHK.
   *============== Routine To Management Errors Condition =====*
        PERFORM E0001-PRT-STATISTICS  THRU E0001-99.
        IF FILE-OPEN
           CLOSE   ZFILE1 ZFILE1O.
        MOVE 99                  TO WS-RET-CODE.
        MOVE WS-RET-CODE         TO RETURN-CODE.
        STOP RUN.
     ERRCHK-EX.
        EXIT.
    E0001-PRT-STATISTICS.
        DISPLAY WS-RIGA-ASTER.
        DISPLAY WS-RIGA-ERR-1.
        DISPLAY '* STATISTICS OF THE PROGRAM ' WS-PGM-ID.
        DISPLAY WS-RIGA-ASTER.
        DISPLAY '* RECORDS   READ                          - '
                                              WS-CTR-READ.
        DISPLAY '* RECORDS   RUN                           - '
                                             WS-CTR-HANDLE.
        DISPLAY '* RECORDS   RUN with ERRORS               - '
                                            WS-CTR-WRITE-A.
        DISPLAY '* TYPE RECORD "N"   WRITE                 - '
                                            WS-CTR-WRISI.
        DISPLAY '* TYPE RECORD NOT   WRITE                 - '
                                            WS-CTR-WRINO.
        DISPLAY WS-RIGA-ASTER.
     E0001-99.
        EXIT.
    C0001-END-JOB.
   *
   * ROUTINE TO MANAGEMENT PRINT STATISTICS, CLOSE FILES AND
   * DEFINE Return Code.
        PERFORM E0001-PRT-STATISTICS  THRU E0001-99.
        DISPLAY WS-RIGA-ASTER.
        DISPLAY '* ' WS-PGM-ID ' - RUN OK |||'.
        DISPLAY WS-RIGA-ASTER.
        MOVE    PLANNUM      TO WS-RET-PNUM.
        MOVE WS-RET-PNUM     TO RETURN-CODE.
```

```
           CLOSE   ZFILE1 ZFILE1O.
           STOP RUN.
         C0001-99.
           EXIT.
```

NvDM SEND PROCEDURE: ISPF PANELS

Panel NSPP000

```
)ATTR
/* NvDM Send Procedure -  Panel NSPP000                              */
$ TYPE(INPUT) INTENS(NON)
 _ TYPE(INPUT)  CAPS(ON)    PAD('_')
)BODY EXPAND (//)
%=/=/= SOFTWARE DISTRIBUTION =/=/=
+
%                     NvDM *** S E N D  P R O C E D U R E
+        ------------------------------------------------------------
+
+                      Insert your%NvDM password $PW      +
+
+      Select option:
+
+     %>_Q%<  Prepare package to send
+
+     %>_R%<+ Display objects of the %Resource Repository+:
+                 Project.......> _PP   + (Alias, high level qualified)
+                 Type..........> _TT   + (SOURCE, LOAD, DBRM, DOC, etc)
+                 Destination...> _D+  (%1+NA1)
+                                      (%2+TR )
+                                      (%3+NA2)
+                                      (%4+MI )
+                                      (%6+RM )
+
+                 Don't fill the fields to display all contents of the
+                 Resource Repository.
)INIT
  .HELP = NSPP000H
  &PW = &PW
  &Q  = &Z
  &R  = &Z
  &PP = &Z
  &TT = &Z
  &D  = &Z
)REINIT
  &PW = &PW
  &Q  = &Z
  &R  = &Z
```

```
)PROC
 VER(&PW,NONBLANK)
  IF (&R ¬= 'S')
   VER(&Q,NONBLANK,LIST,S)
  IF (&Q ¬= 'S')
   VER(&R,NONBLANK,LIST,S)
  IF (&R = 'S')
   IF (&D ¬= ' ')
    VER(&D,NONBLANK,LIST,1,2,3,4,6)
)END
```

Panel NSPP000H

```
%---------------------- NvDM  ***  SEND PROCEDURE ----------------------

+               ----------------------------------------
%                                INFO
+               ----------------------------------------

This procedure carries out the software packing and the distribution of
the packing to other subarea hosts with the NetView Distribution Manager
product.
It is a simple procedure to use.
The procedure transparently creates and executes the NvDM utilities. The
customer must only select the datasets/files to send and the relative
destinations.
Its main functions are:
%-+display the content of the Resource Repository;
%-+select the files to send;
%-+select nodes NvDM destinations;
%-+prepare packages (function NvDM Maintdat/prepare);
%-+prepare the list of the objects to send (function NvDM Tcfmaint/
plan);
%-+execute the NvDM utilities;
%-+control the outcome of the NvDM utilities;
%-+start the transmission of the software selected towards NvDM nodes
chosen;
%-+supply log of all activities carried out;
)END
```

Panel NSPP001

```
)ATTR
/* NvDM Send Procedure  -  Panel NSPPØØ1
*/
¬ TYPE(INPUT) INTENS(LOW) PAD(_)
)BODY EXPAND ($$)
%=$=$= SOFTWARE DISTRIBUTION =$=$=
%COMMAND ===>_ZCMD
```

```
+
%-$-$- NvDM *** S E N D  P R O C E D U R E -$-$-
%
+ISPF Library:                                          %* * * * * * *
* * *+
+  Project    %===>_UNO     +                           %*+ABORT(Y/N)?
_O% *+
+  Group      %===>_DUE     +                           %* * * * * * *
* * *+
+  Type       %===>_TRE     +
+  Member     %===>_MEM     +  (Blank or pattern for member selection
list)
+
+SEQUENTIAL DATA SET:
+   DATASET NAME  %===>_DAS
+
+
%-$-$- NvDM node destination -$-$-
+                       (Select with "S" the destination)
+
+      _X+ ...<ALL DEST>      _W+ ....NAPOLI 2       _V+ .....NAPOLI 1
+
+      _Q+ ....MILANO         _S+ ....PALERMO        _U+ .....ROMA
+
+ %PF3+end selection
)INIT
  .HELP = NSPPØØØH
  .CURSOR = UNO
  &ZCMD = &Z
  &UNO = &UNO
  &DUE = &DUE
  &TRE = &TRE
  &MEM = &Z
  &DAS = &Z
  &O = 'N'
  &Q = '_'
  &U = '_'
  &V = '_'
  &W = '_'
  &S = '_'
  &X = '_'
)REINIT
  &ZCMD = &Z
  &UNO = &UNO
  &DUE = &DUE
  &TRE = &TRE
  &MEM = &MEM
  &DAS = &DAS
  &O = &O
  &Q = &Q
```

```
  &U = &U
  &V = &V
  &W = &W
  &S = &S
  &X = &X
)PROC
  IF (&O ¬= 'Y')
   IF (&Q ¬= 'S')
    IF (&U ¬= 'S')
     IF (&V ¬= 'S')
      IF (&W ¬= 'S')
       IF (&S ¬= 'S')
        VER(&X,NONBLANK)
        VER(&X,LIST,S)
   IF (&DAS = ' ')
      VER(&UNO,NONBLANK)
      VER(&DUE,NONBLANK)
      VER(&TRE,NONBLANK)
)END
```

Panel NSPP002

```
)ATTR
/* NvDM Send Procedure  -  Panel NSPPØØ2
*/
@ TYPE(OUTPUT) INTENS(LOW)
 ¬ TYPE(OUTPUT) INTENS(HIGH)
 _ TYPE(INPUT)  CAPS(ON)
)BODY EXPAND (//)
%=/=/= SOFTWARE DISTRIBUTION =/=/=
+COMMAND%===>_ZCMD
+
%-/-/- Members selection -/-/-
    %S+to select member    %ALL+on COMMAND line to select all members
+ %PF3+end selection
+
+            S    MEMBER LIST    of  ¬NOMM
+
%           ---   -----------  +
)MODEL
            _Z+    @Z         +  ¬Z         +
)INIT
  .HELP = NSPPØØØH
  .CURSOR = &ZCMD
  &MS = &Z
  &ZCMD = &Z
  .ZVARS='(MS MNAME SELE)'
)REINIT
  REFRESH(MS)
```

```
)PROC
 IF (&ZTDSELS ¬= 0000)
    VER(&MS,LIST,S)
)END
```

Panel NSPP003

```
)ATTR
/* NvDM Send Procedure -  Panel NSPP003
*/
@ TYPE(OUTPUT) INTENS(HIGH)
 $ TYPE(INPUT) INTENS(NON)
 _ TYPE(INPUT)  CAPS(ON)
)BODY EXPAND (//)
%=/=/= SOFTWARE DISTRIBUTION =/=/=
+COMMAND%===>_ZCMD
+
%-/-/- DISTRIBUTIONS SELECTED -/-/-
%OPTIONS: +Insert%D+near distributions to delete
          +Insert%ALL+on Command line to delete all distributions
+          %PF3+to send distributions
+
+D  Object of the distribution        NvDM Node destination
)MODEL
_Z @Z                                 @Z @Z    @Z  @Z  @Z @Z    @Z  @Z
)INIT
  .HELP = NSPP000H
  &DS = &Z
  &ZCMD = &Z
  .ZVARS='(DS DOBJ DTR DMI DRM DPA DFI DGE DNA1 DNA2)'
)REINIT
  &DS = &Z
  REFRESH(DS)
)PROC
 IF (&ZCMD ¬= 'ALL')
   IF (&ZTDSELS ¬= 0000)
     VER(&DS,LIST,D)
)END
```

Panel NSPP004

```
)ATTR
/* NvDM Send Procedure  -  Panel NSPP004
*/
 @ TYPE(OUTPUT) INTENS(HIGH)
 $ TYPE(OUTPUT) INTENS(HIGH)  JUST(RIGHT)
)BODY EXPAND (//)
%=/=/= SOFTWARE DISTRIBUTION =/=/=
+COMMAND ===>_ZCMD
```

```
+
%-/-/- Contents of the RESOURCE REPOSITORY -/-/-
+
% PF7+Up                        %PF8+Down                          %PF3+End
+
)MODEL
+-------------------------------------------------------------------------
--------
$TN  +  Object name  @TNAM
+
+       Destination :@TDES       + Into RR : @TDAT    +@TTIM +
)INIT
  .HELP = NSPPØØØH
  .CURSOR = &ZCMD
  &ZCMD = &Z
)END
```

Espedito Morvillo
(Italy)

Free weekly news by e-mail

Xephon has four weekly news services covering the following subject areas:

- Data centre
- Distributed systems
- Networks
- Software.

Each week, subscribers receive, by e-mail, a short news bulletin consisting of a list of items; each item has a link to the page on our Web site that contains the corresponding article. Each news bulletin also carries links to the main industry news stories of the week.

To subscribe to one or more of these news services, or review recent articles, point your browser at http://www.xephon.com/news.htm.

Printing from CICS to an IP printer

Our problem: how to print from CICS to an IP printer. Our solution: NetSpool and IP PrintWay.

NetSpool converts data received from VTAM applications into System/370 line data and places the data on the JES2 or JES3 spool. After NetSpool has created an output dataset on the JES spool, JES or PSF/MVS can print the dataset or transmit it to another location for printing. NetSpool can be configured for your installation so that you don't need to change existing VTAM applications. That is, existing VTAM applications send print requests to NetSpool in the same way as they currently send print requests to SNA-network printers.

By placing VTAM application output on the JES2 or JES3 spool, NetSpool enables you to use the security, checkpoint/restart, and reprint capabilities provided by JES.

To facilitate the use of NetSpool with IP PrintWay, you can define NetSpool logical printers in the routing and options datasets used by IP PrintWay, instead of in the NetSpool print-characteristics dataset.

IP PrintWay transmits output datasets from the JES spool to printers in a TCP/IP network.

PRINTSERVER AND IP/PRINTWAY

The Network Print Facility lets you print data from your MVS/SP system on remote printers accessible through IBM TCP/IP for MVS. The Network Print Facility supports the printing of the following types of output:

- JES2 output.
- JES3 output.
- VTAM SNA character string (SCS) output over LU type 1 sessions.
- VTAM 3270 datastream output over LU type 3 and LU type 0 sessions.

It accomplishes this by transforming VTAM or JES output print data into a format that existing LPD functions can process.

The Network Print Facility lets you decide where and how output will be printed through the use of a routing file and options file. The 'where' portion is defined by the routing file. The 'how' portion is defined through LPR options in the options file and through other data in the routing file.

The Network Print Facility allows user exits for installation-defined routing decisions or data modifications. It also provides a queue manager program to control the initial sending of each print job, retries of failed print jobs, and deletion of print datasets after a user-specified retention time.

PROC AND ISPF PANELS

The PRINTSRV procedure has only one line:

```
//STEP EXEC PGM=BPXBATCH,PARM='SH aopstart'
```

- Modify the TSO log-on procedure for print administrators.
- Add in SYSEXEC: SYS1.SANFEXEC.
- Add in ISPLLIB: SYS1.SAOPLOAD and SYS1.SAN FLOAD .
- Add in ISPMLIB: SYS1.SAOPMENU and SYS1.SAN FMLIB.
- Add in ISPPLIB: SYS1.SAOPPENU and SYS1.SANFPLIB.
- Use the ISR$390S panel or update your own panels as follows:
  - Print Server: CMD(AOPINIT &ZTRAIL) NEWAPPL(AOP) SCRNAME(PRTINT) NOCHECK.
  - IP/Printway: PANEL(ANFIPM) SCRNAME (IPPRINT).

PRINT SERVER

- Define the RACF FACILITY AOPADMIN resource with READ access for users.

- Define the RACF STARTED PRINTSRV.* with link on the user PRINTSRV (with uid(0) home(/usr/lpp/Printsrv/bin) program(/bin/sh)).
- Define the RACF FACILITY BPX.FILEATTR.APF resource with Read access for administrators.
- Start the following command:

```
/usr/lpp/Printsrv/bin/aopsetup <grp_exp> <grp_systeme>
```

- Check RACF for printers (WRITER class) in the ISPF panel printsrv and add your printers.

UNDER WINDOWS 9X OR NT

- With ftp in bin mode, download the following files from /usr/lpp/Printsrv/win/en_US:
  - afpdrv95.exe (AFP for Windows 90*x*)
  - afpdrvnt.exe (AFP for Windows NT)
  - afpviewr.exe (‘plug-in’ of Internet Explorer or Netscape Navigator)
  - aopwin.exe (Ports for Print Server).
- Complete pilot installation.
- Port administration:
  - Execute: aopwin <rep_pour_aopwin>
  - Execute from <rep_pour_aopwin>: setup
  - Follow instructions
  - Restart Windows 9*x*.

PRINTER DRIVER AFP

- Execute: afpdrv95 <rep_for_afpdrv95> (complete branch).
- Do Add printer local and:

– Choose <rep_for_afpdrv95> to search the pilots
– Choose model AFP 300
– Choose port OS/390 and update the ip address.

The following printers work well:

- Siemens 2050
- MDS 90/180
- IBM 4332
- AFP IBM 3827
- AFP IBM 3825
- AFP 300.

IP PRINTWAY

IP Printway is a subsystem for JES2 like PSF/MVS. IP Printway should be defined like a fss and as a (PRTxx)

Installing IP PRINTWAY

ANFDEAL

```
//PSY3ALLC JOB SYS,SYSTEMS,CLASS=S,NOTIFY=PSY3,MSGCLASS=T
//*  ALLOC PRINTWAY FILES
//DEL      EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
       DELETE -
            TCPIP.ANF.ROUTING
       DELETE -
            TCPIP.ANF.OPTIONS
       DELETE -
            TCPIP.ANF.QUEUE
//CREATE   EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN    DD   *
   DEFINE CLUSTER -
             (NAME(TCPIP.ANF.ROUTING) -
                INDEXED -
                SPEED -
```

```
                SHAREOPTIONS(3 3)) -
              DATA -
                (NAME(TCPIP.ANF.ROUTING.DATA) -
                TRACKS(3Ø 5) -
                KEYS(21 Ø) -
                RECORDSIZE(13Ø 628) -
                FREESPACE(1Ø,1Ø) -
                CISZ(4Ø96)) -
              INDEX -
                (NAME(TCPIP.ANF.ROUTING.INDEX) -
                IMBED)

   DEFINE CLUSTER -
            (NAME(TCPIP.ANF.OPTIONS) -
                INDEXED -
                SPEED -
                SHAREOPTIONS(3 3)) -
              DATA -
                (NAME(TCPIP.ANF.OPTIONS.DATA) -
                TRACKS(3Ø 5) -
                KEYS(16 Ø) -
                RECORDSIZE(56 568) -
                FREESPACE(1Ø,1Ø) -
                CISZ(4Ø96)) -
              INDEX -
                (NAME(TCPIP.ANF.OPTIONS.INDEX) -
                IMBED)

   DEFINE CLUSTER -
            (NAME(TCPIP.ANF.QUEUE) -
                INDEXED -
                SPEED -
                SHAREOPTIONS(4 3)) -
              DATA -
                (NAME(TCPIP.ANF.QUEUE.DATA) -
                TRACKS(3Ø 5) -
                KEYS(2Ø Ø) -
                RECORDSIZE(238 992) -
                FREESPACE(1Ø,1Ø) -
                CISZ(4Ø96)) -
              INDEX -
                (NAME(TCPIP.ANF.QUEUE.INDEX) -
                IMBED)

/*
```

ANFIROQ

```
//ASM     EXEC PGM=ASMA9Ø,PARM='OBJECT,TERM',REGION=1Ø24K,COND=EVEN
//SYSLIB   DD DSN=SYS1.SANFMAC,DISP=SHR
```

```
//         DD DSN=SYS1.MACLIB,DISP=SHR
//SYSUT1   DD UNIT=SYSDA,SPACE=(CYL,(5,1))
//SYSUT2   DD UNIT=SYSDA,SPACE=(CYL,(5,1))
//SYSUT3   DD UNIT=SYSDA,SPACE=(CYL,(5,1))
//SYSPUNCH DD DUMMY
//SYSLIN   DD DSN=&&OBJSET,DISP=(MOD,PASS),UNIT=SYSDA,
//            SPACE=(80,(500,50))
//SYSTERM  DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
*
*   Initialize VSAM datasets
*
*
         ANFGPWFL TYPE=INITIAL,MODE=LOAD,PRGNAME=ANFPWVSU
*
*   Final statement
*
         ANFGPWFL TYPE=FINAL
/*
//*  LINK THE FILE LOAD PROGRAM
//LKED    EXEC PGM=IEWL,PARM='LIST,MAP,XREF',REGION=512K,COND=(4,LT)
//SYSPRINT DD SYSOUT=*
//SYSUT1   DD UNIT=SYSDA,SPACE=(CYL,(5,1))
//SYSLMOD  DD DSN=&&GOSET(GO),DISP=(MOD,PASS),UNIT=SYSDA,
//            SPACE=(1024,(50,1,1))
//SYSLIN   DD DSN=&&OBJSET,DISP=(OLD,DELETE,DELETE)
//LINK.SYSIN DD DUMMY
//*  EXECUTE THE FILE
//LOAD    EXEC PGM=*.LKED.SYSLMOD,REGION=512K,COND=(4,LT)
//ANFROUTG DD DSN=TCPIP.ANF.ROUTING,DISP=OLD
//ANFOPTNS DD DSN=TCPIP.ANF.OPTIONS,DISP=OLD
//ANFQUEUE DD DSN=TCPIP.ANF.QUEUE,DISP=OLD
//SYSPRINT DD SYSOUT=*
//SYSABEND DD SYSOUT=*
//SYSMDUMP DD SYSOUT=*
//
```

ANFMIJCL

```
//*-------------------------------------------------------------------*//
//* INITIALIZE THE IP PRINTWAY MESSAGE QUEUE FILE.                    *//
//*-------------------------------------------------------------------*//
//REXXJCL  EXEC PGM=IRXJCL,PARM='ANFMFILE'

//SYSTSPRT DD  SYSOUT=*
//SYSEXEC  DD  DSN=SYS1.SANFEXEC,DISP=SHR
//MSGDD    DD  DSN=TCPIP.ANF.MSGFILE,DISP=(NEW,CATLG,DELETE),
```

```
//             UNIT=SYSDA,SPACE=(CYL,1),
//             DCB=(DSORG=PS,RECFM=FB,LRECL=Ø8Ø,BLKSIZE=Ø8ØØ)
/*
```

Modifying JES2 parameters

```
FSSDEF(PRINTWAY) PROC=ANFWPROC
PRINTER7 FSS=PRINTWAY,
       PRESELCT=NO,
       PRMODE=(LINE,PAGE,SOSI1),
       ROUTECDE=(PRTHP6),
       CLASS=1,
       UCS=Ø,
       SEP,
       NOSEPDS,
       CKPTPAGE=1Ø,
       DRAIN,
       MARK
```

ANFWPROC procedure

```
//ANFWPROC PROC  HLQ='TCPIP.ANF'
//* ------------------------------------------------
//* THIS IS A SAMPLE PROCEDURE FOR STARTING IP PRINTWAY
//* ------------------------------------------------
//IEFPROC  EXEC  PGM=ANFFIEP,REGION=8M,TIME=NOLIMIT,PARM=(8ØØØ)
//*-----------------------------------------------------
//*  NOTE: THE PARM=(8ØØØ) SPECIFIES THE DEFAULT HIPERSPACE
//*        ALLOCATION FOR EACH FSA, WHICH IS 32M
//*        (8ØØØ-4K BLOCKS).
//*        WARNING: THIS IS ALLOCATED FROM THE SYSTEM
//*                 PAGE SPACE AND MAY AFFECT SYSTEM
//*                 PERFORMANCE IF THE SIZE IS TOO LARGE
//*                 OR MANY FSAS ARE RUNNING.
//*-----------------------------------------------------
//STEPLIB  DD DSN=SYS1.SANFLOAD,DISP=SHR
//         DD DSN=SYS1.SCEERUN,DISP=SHR
//         DD DSN=SYS1.SEZALINK,DISP=SHR
//ANFROUTG DD DSN=&HLQ..ROUTING,DISP=SHR
//ANFOPTNS DD DSN=&HLQ..OPTIONS,DISP=SHR
//ANFQUEUE DD DSN=&HLQ..QUEUE,DISP=SHR
//ANFMMSG  DD DSN=&HLQ..MSGFILE,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSABEND DD SYSOUT=*
//SYSMDUMP DD SYSOUT=*
//PRINTER7 CNTL
//PRINTER7 PRINTDEV TRACE=NO
//PRINTER7 ENDCNTL
```

ISPF

In the IP/Printway panel under ISPF, choose add route.

RACF

The RACF definition is as follows: RACF STARTED ANFWPROC

Claude Dunand
(France)

Call for papers

Why not share your expertise and earn money at the same time?

TCP/SNA Update is looking for REXX EXECs, macros, CLISTs, program code, etc, that experienced networkers have written to make their lives, or the lives of their users, easier. We will publish it (after vetting by our expert panel) and send you a cheque when the article is published. Articles can be of any length and can be sent or e-mailed to Fiona Hewitt at any of the addresses shown on page 2.

More information about how to contribute is contained in our free booklet entitled *Notes for Contributors*. Please contact us, and we will send you a copy. Alternatively, you can download it from our Web site, at:

www.xephon.com/contnote.html.

Web-to-host integration in practice

Browser-based Web-to-host integration, which came into being in mid-1996 with the introduction of the first 3270-to-HTML conversion product developed by Tuebner Associates (now Esker), can no longer be thought of as an unproven technology. Despite being negatively impacted by Year 2000-related activities, Web-to-host integration initiatives have been proceeding at a respectable pace over the last couple of years, and well over 1,500 corporations worldwide have already successfully adopted the technology. Some of the 'blue-chip' companies that are currently using Web-to-host – and, in particular, Web-to-mainframe – technology include: General Motors, FedEx, American Airlines, Continental Airlines, Florida State University (FSU), Trans World Airlines (TWA), Bank of America, Charles Schwab, Nestlé, Lafayette Life Insurance, Gerling Group (Germany), and Del Monte Foods. Charles Schwab, the world's largest discount brokerage, now does over 60% of its trade on-line, across the Internet, using Web-to-host technology coupled to six IBM mainframes.

This article describes three very different real-life case studies, which can be summarized as follows (see also Figure 1):

- Continental Airlines, the fifth largest airline in the US, offers over 2,200 flights daily to 136 domestic and 87 international destinations. In the context of Web-to-host, Continental provides a textbook example of how secure Web-to-host integration over the Internet can be exploited to provide mobile users (the flight crews) with easy and inexpensive access to mainframe applications.
- The Charleston County Court, a public court system in South Carolina, USA, shows the other dimension of Internet-based host access – that of providing the general public with the ability to easily interact with host applications. Obviously, the harsh, anachronistic 'green-on-black' interface of a typical 3270 application is not conducive to use by the general public. Instead, the interface to the mainframe applications needs to be rejuvenated, ideally with a contemporary 'point-and-click' GUI, so that it is intuitive, welcoming, and forgiving.

Customer	*Industry sector*	*Web-to-host technology*	*Key payoff*	*Solution provided by*	*Rejuvenated user interface*
Continental Airlines	Transport	3270-to-HTML and applet-based tn3270(E) emulation	Economical mainframe access over the Internet from around the world	Farabi Technology	Yes
Charleston County Court	Local govt	3270-to-HTML	Public access to mainframe applications	iE	Yes
Bank of America	Banking	Java-based applets and servlets	Significantly extending the scope of a mainframe application without making any changes to the mainframe code	ResQNet.com	Yes

Figure 1: Summary of case studies

- Bank of America, the largest bank in the US, demonstrates how off-the-shelf Web-to-host technology can be extended through the use of Java servlets to significantly enhance the scope and applicability of existing mainframe applications.

CONTINENTAL AIRLINES

Thanks to Web-to-host technology, Continental Airlines now lets its flight crew personnel bid, on-line across the Web, for the flight schedules (ie their work roster in terms of flights flown) they would like to fly during forthcoming work cycles. A mainframe-based flight crew scheduling application then produces flight schedules that accommodate as many of the preferences as possible, using seniority and other credentials (eg language proficiency relative to flight destinations) to decide precedence.

Flight crew personnel work relatively long and variable hours for three to four days and then typically have three to four days off – inevitably with some degree of variability within this overall cycle. The bidding scheme offers a direct and impartial mechanism to influence their work roster – and saves having to continually seek intervention from supervisors or swap schedules with friends.

Scheduling thousands of flight crew personnel to handle thousands of flights around the world, each week, is logistically highly complex, especially with having to factor in possible contingencies such as the impact of flight delays caused by weather or air traffic, or the maximum permissible hours a given crew can work. Airlines use highly sophisticated programs, invariably mainframe-based, which have been refined and fine-tuned over the decades to work out their flight schedules. The flight crew scheduling application is as important and mission-critical as the flight reservation system – without flight crew there can be no flights!

Providing universal and easy on-line access to this mainframe-based bidding process was obviously essential if the scheme was to succeed and, above all, to be deemed equitable by the participants. Web-to-host integration, with 3270-to-HTML conversion, was the obvious solution. With a totally browser-centric, 3270-to-HTML conversion scheme, flight crew personnel would be able to participate in the

bidding process from home or a hotel as long as they had access to a PC, workstation, or laptop with a standard browser and dial-up capability. With access to an appropriate global ISP, they could reach the mainframe, effortlessly, from anywhere in the world.

Since there's no reliance on applets with a standard 3270-to-HTML scheme, the mainframe access can be realized with impunity from any platform, be it an iMac or PC running Linux, whether it's borrowed from a friend or time-shared from an Internet cafe in Beijing, without the delay of an applet download or the increasing platform-dependent vagaries of current applet-based emulators. Therein lies the appeal and power of 3270/5250-to-HTML-based host access.

Continental chose Farabi Technology's HostFront Publishing (see www.farabi.com). HostFront Publishing is an NT-based 3270-to-HTML conversion solution that offers extensive rejuvenation capabilities through Microsoft's Active Server Page (ASP) based scripting (which includes the use of Microsoft's VBScript) or a straightforward and intuitive Farabi Script Language (FSL).

HostFront Publishing is in effect the 3270/5250-to-HTML conversion feature of the overall HostFront Web-to-host access solution which also includes 3270/5250 emulation, now complete with an AutoGUI capability, using Java or ActiveX emulators. HostFront is a quintessential 'three-tier' Web-to-host access scheme, reminiscent of OpenConnect's once market-leading OC://WebConnectPro – with the NT-based HostFront Server gateway component always a prerequisite whether applet-based emulation or 3270/5250-to-HTML is being used for Web-to-host access.

In the old days, the HostFront Server could obtain SNA connectivity only by talking, via an API, to a co-resident Microsoft SNA Server gateway. Now, however, HostFront Server can also act as a *bona fide* 'tn' client and gain SNA connectivity through any standard tn3270(E)/tn5250 Server – which also does not have to be on the same box as the HostFront Server.

HostFront provides two separate security functions:

- Optional, user-id/password-based user authentication for both applet-based and HTML-based access.

- Applet-to-server data encryption via a proprietary protocol that uses a variable encryption key that changes between each transaction.

HostFront Server therefore has two roles. The first is to obtain SNA connectivity either via the SNA Server API or using 'tn' protocols. The other is to provide security services – with the optional user authentication function usable even when HostFront Publishing is being used – as is the case with Continental. Although not used by Continental, HostFront Publishing also has a Macromedia Shockwave capability that allows full-motion graphics to be interspersed with host output. Continental could think about using this feature to convey any breaking news to their staff.

Continental's flight scheduling application is outsourced to EDS and runs at an EDS mainframe data centre in Charlotte, North Carolina. Continental's operational headquarters is in Houston, Texas. Connectivity between the NT servers running the HostFront software and SNA Server and the data centre in Charlotte is via leased SDLC links which are a part of the whole outsourcing agreement.

HostFront Server and HostFront Publishing have to be co-resident on the same NT server. HostFront Publishing also requires a co-resident Web server – which in most instances, including at Continental, is Microsoft's ubiquitous Internet Information Server (IIS). Continental has opted for the co-resident, API-based SNA Server configuration, although this is no longer mandatory.

HostFront Server used to have a concurrent session limit of 128. This is no longer the case, however, and it can now support up to 1,024 concurrent sessions in the case of mainframe access and 512 in the case of AS/400s (depending on the power of the NT server).

Continental currently uses around 900 concurrent sessions, spread across seven servers running HostFront software. Load-balancing and fault-tolerance is realized in this seven server configuration using a combination of Microsoft's Windows NT Load Balancing (WLBS) and a router-based capability customized by Continental. In the current configuration, each server is handling around 128 concurrent sessions as per the old limitations. With the new limits, Continental could easily handle three times the volume with this same seven server set-up.

The flight schedule bidding system uses customized screens – which obviously appear as Web pages. Once the system was successfully implemented, Continental realized that it could use the already installed Farabi software to provide some of its in-house corporate users with 'green-screen' emulation-based mainframe access. Farabi was one of the first vendors to support both Java and ActiveX emulators. (The other vendors that offer both types of emulator are Attachmate, NetManage, ICOM, and Esker.)

This case study offers a great example of Web-to-host technology in action with a high-profile customer. Airlines are prime candidates for Web-to-host integration, and most already have some level of Web-to-host capability – with some, such as American Airlines, offering full-service travel reservation with credit card billing. This case study also illustrates the advantages of using HTML conversion and applet-based emulation in tandem – the former for Internet-based access and the latter for in-house corporate access with 'green-screen' emulation.

CHARLESTON COUNTY COURT

The experiences of Charleston County Court highlight how today's Web-to-host integration technology – and in particular 3270-to-HTML conversion, which is the 'thinnest' of 'thin client' solutions – can be effectively and profitably used to offer the general public authorized access to public records.

The Court of Common Pleas and General Sessions for Charleston County in South Carolina takes place at the Charleston County Judicial Center, located in North Charleston, SC. This court is a part of the Circuit Court of SC – where Charleston county in conjunction with Berkely county form the 9th Judicial Circuit of the state's 16 circuits.

The Court of Common Pleas is the civil branch of the circuit court and has jurisdiction over all civil cases in Charleston where the amount in dispute is in excess of $5,000. The Court of General Session handles criminal cases – excepting most misdemeanour cases, which are usually heard in the Magistrates' Courts.

Case records of the Courts of Common Pleas and General Session, as well as those of the Small Claims Courts and out-of-county judgements,

are deemed to be public domain information that the general public has a right to access and peruse. The Clerk of Court is responsible for maintaining and managing these records.

Up until mid-1999, the general public, paralegals, or lawyers who required access to Charleston County legal documents or case records had to visit one of two county offices. This was obviously not very convenient.

Earlier this year, however, the Charleston County Clerk of Court decided to provide access to court documents over the Web (www3.charlestoncounty.org). Since the case records were maintained on a mainframe, Charleston County Court needed a solid Web-to-host solution, and chose Intelligent Environments' ScreenSurfer.

ScreenSurfer performs 3270/5250-to-HTML conversion on-the-fly and also offers extensive user interface rejuvenation. Its features include:

- Session integrity (ie 'persistence') and session time-out control through the use of cookies.
- Function key support through an ActiveX 'applet'.
- One-step, light-pen selection through the use of 'clickable' Graphical Interchange Format (GIF) tags.
- Integrated Web server.
- Support for Microsoft Active Server Page (ASP) technology for enhancing Web pages.
- Optional user authentication.
- Correct alignment of all host-generated columns and tables.
- Support for at least 1,000 concurrent sessions when installed on a typical NT-server configuration through extensive use of multithreading.

Given this feature set, ScreenSurfer credibly competes with Novell's HostPublisher and Eicon's Aviva Web-to-Host Server for top honours in this arena – with the only caveat being that ScreenSurfer relies on ActiveX for some of its value-adds, thus becoming Windows-specific

in some instances, whereas the other two use Java for their value-adds to realize platform independence.

The Charleston County MIS Department was able to develop and deploy the ScreenSurfer-based Web-to-host access in just eight weeks – and that included the extensive rejuvenation of quite a few mainframe screens. Today, ScreenSurfer is handling around 2,800 mainframe access requests a day.

BANK OF AMERICA

The experiences of Bank of America offer an inspiring case study of how standard, Java-based, Web-to-Host technology can be easily leveraged through macros, scripts, and auxiliary Java servlets to totally alter the complexion, flow, and scope of an existing mainframe application.

In this instance, the application was Bank of America's in-house supplies-ordering application, which is CICS based and is known as OEXR. The Java technology to streamline and enhance the application was provided by ResQNet and was centred around a variant of the standard applet it uses to 'face-lift' IBM's Host On-Demand.

The overall architecture of the ResQNet solution and the data flows involved are shown in Figure 2. Note the SmartStream database system from Sybase running on an IBM RS/6000. Being able to readily access data from this database to complement the information available through the CICS OEXR application is a key feature of this solution.

ResQNet developed a servlet to work in conjunction with its client-side applet to realize this data augmentation function. This servlet also produces an HTML version of the order being placed so that it can be presented, electronically, in browser-viewable form to a designated manager. The servlet (as shown in Figure 2) runs on another RS/6000 system – the same one that hosts the Lotus Domino GO Web Server being used by Bank of America for this nationwide intranet application.

OEXR is in effect an electronic, on-line supplies cupboard enabling authorized Bank of America employees around the country to requisition supplies, including commodity hardware and networking

products. OEXR currently lists about 4,000 individual items which include obvious staples such as pens, staplers, and three-ring punches, as well as capital-cost items such as PCs. OEXR is a normal mainframe-resident CICS application, and therefore needs to be accessed using SNA/3270. Bank of America uses the Host On-Demand Java applet, downloaded from the Lotus Domino GO Web Server (and subsequently cached), to achieve this connectivity across its intranet – using a mainframe resident tn3270(E) Server.

One of the goals in updating, streamlining, and enhancing the OEXR order processing scheme was to give users a contemporary, mouse-navigatable, point-and-click GUI in place of the dated 'green-and-black' 3270 screens.

ResQNet does all of its rejuvenation at the client using a Java applet. The AutoGUI, or default transformations, are built into this relatively small (< 500K) applet. Custom changes, including screen consolidation scenarios, developed using ResQNet's Customization Studio are defined and downloaded to the client, for execution by the standard ResQNet applet, in the form of an XML (Extensible Mark-up Language) file.

Rejuvenating the user interface to meet the Bank of America requirements using standard ResQNet technology was fairly straightforward. What's interesting about this case study is the extension of the standard ResQNet applet, in conjunction with a servlet, to augment the CICS application via data that was available only on the SmartStream database system.

The two primary enhancements sought by Bank of America for this in-house supply ordering system, over and above rejuvenating the 3270 user interface, were to:

- Automate and expedite the order approval and processing cycle by converting the order for supplies made by a Bank of America user into a properly formatted and printable purchase order document which could then be routed to a designated manager authorized to approve such an order.
- Enable users to check the status of previously placed orders for supplies – on-line.

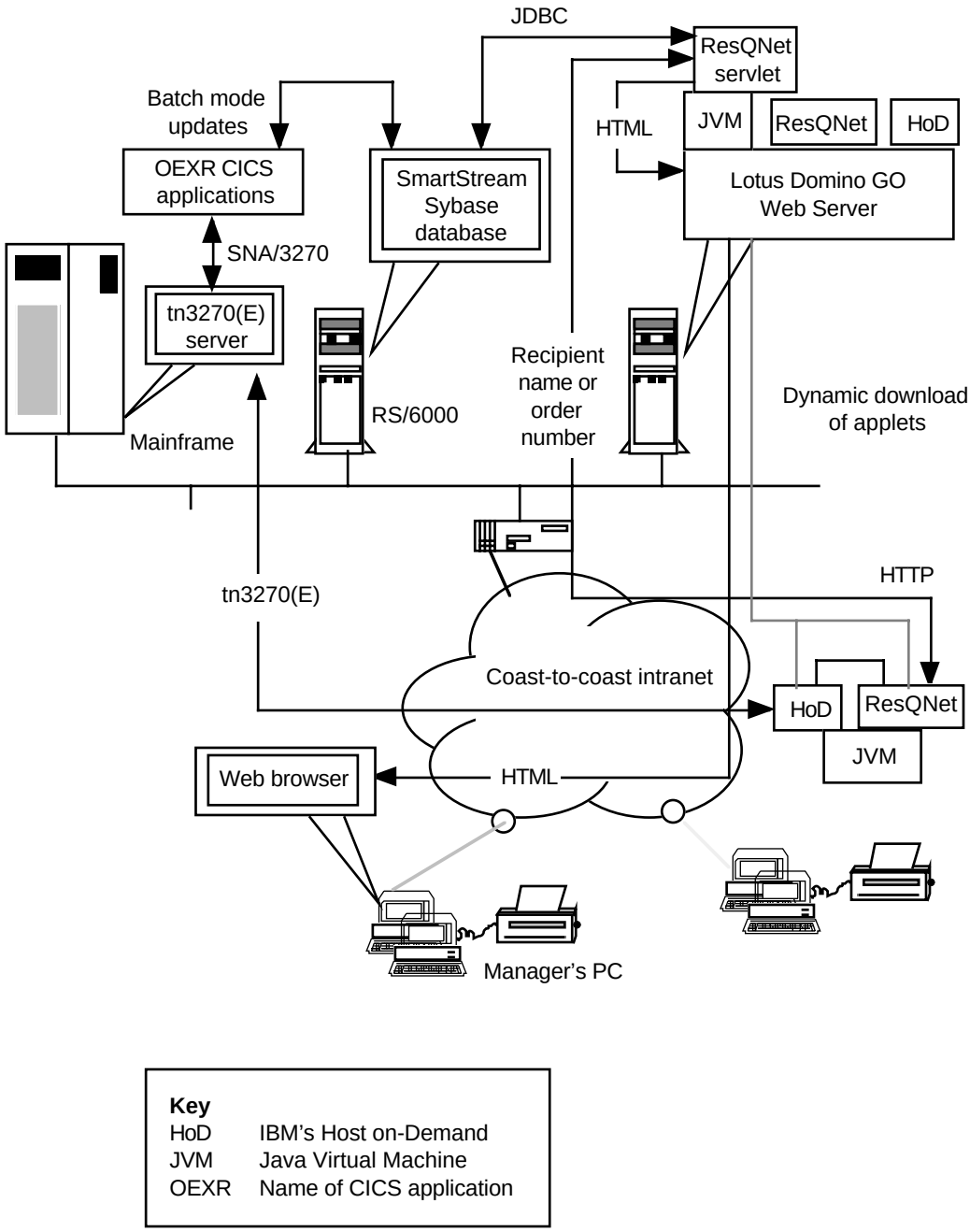

Figure 2: The architecture and general data flow of the automated Bank of America in-house supplies ordering system

The only way to realize these enhancements was to combine information from the OEXR CICS application with data found in the SmartStream database. ResQNet developed a servlet (as shown in Figure 2) to retrieve the pertinent information from SmartStream using Java Database Connectivity (JDBC). The ResQNet applet that does on-the-fly rejuvenation for the Host On-Demand applet (via the Host On-Demand Host Access Class Library (HACL) interface) was modified to talk to the ResQNet servlet – as well as performing the rejuvenation functions required by Bank of America. HTTP is used for all of the communications between the ResQNet applet and the servlet. HTTP's standard GET and PUT requests are used to achieve the requisite data transfers between these two Java-based components.

A Bank of America employee would set about placing an order for supplies by invoking the CICS OEXR application. This is now done using Host On-Demand connectivity coupled to ResQNet's rejuvenation function. The user will then select the items required from the list of over 4,000 products maintained by OEXR. Given the size of this list, a user will typically have to navigate through multiple screens to locate all of the necessary items. During this item selection process, a custom script running on the ResQNet applet automatically collects information on the items being ordered – in parallel to and independent of the OEXR data transfers to and from the mainframe.

Once the user has completed the order, the ResQNet applet contacts the ResQNet servlet, via HTTP, and conveys to it the list of items ordered. The servlet accepts this list and constructs an order form in the form of an HTML Web page. The order form in HTML format is then stored on the Web server. The HTML Web page contains information from the OEXR system as well as data culled from the SmartStream database. The OEXR does not contain the actual shipping address. Instead, it has a shipping code. The servlet uses the OEXR code as the JDBC key to retrieve the shipping address from SmartStream.

Designated managers can retrieve and view the HTML-based order forms (and any subsequent changes made to an order) stored at the Web server using a standard Web browser. Users can also check the status of previously placed orders using this system.

The ResQNet servlet uses JDBC, with the order number as the key, to

obtain the order status information from SmartStream. The information obtained is then formatted by the servlet to create a Web page. This Web page is then forwarded to the user to be displayed within the previously opened browser window.

CONCLUSION

Web-to-host integration technology should no longer be thought of as an emerging technology. Instead, Web-to-host is already highly proven and, as demonstrated by these case studies, is already being actively used in mission-critical scenarios.

Anura Gurugé
Strategic Analyst (USA)

Information point – reviews

Unlike previous articles in this series, this issue is dedicated to a single topic: finding information that will help you make decisions on the PC-based workstation and server products that communicate to the mainframe. Because these hardware and software products run on PCs, they show up on the radar screen of the mega-subscriber media moguls.

We look at three large sites that provide information on a lot of other areas than just TCP/IP and SNA communications, so you'll find them useful for other areas as well. Although our focus here is on selecting the right product, there's a lot of great information available on these sites covering the technology itself, current and future directions, alternative technologies, and management approaches.

THE BIG THREE

There are others, of course, but ZDNet, CNET, and IDG.net are the most comprehensive Web sites for independently-researched information on PC hardware and software. All three are free, and allow you to look up the information you'd normally need without setting up an ID and password. Each has its own unique history.

ZDNet

The ZD in ZDNet stands for Ziff-Davis, best known for magazines like *PC Magazine* and *eWeek* (probably better known by its previous name of *PC Week*). In terms of reliability, I've long rated *PC Magazine* second only to *Consumer Reports* as a source of affordable reviews. While they may not be non-commercial, they do spend the money in their PC Labs division to do a thorough job of testing.

However, things got really confusing last April, when ZD sold everything but its Internet and Events businesses. The most confusing part is that the new owners of the publishing business are calling themselves Ziff Davis Media, without the hyphen. And the original Ziff-Davis is spinning off ZD Events and renaming it Key3Media Group, and naming the remaining Internet company ZDNet.

Despite the sale, ZDNet still carries the content of the ZD magazines. And that's where you'll find most of the value here.

CNET

Starting from scratch in December 1992, CNET began the *CNET Central* television series in April 1995 and CNET.com in June of the same year. Additional television shows and Web sites followed, including Snap.com, in partnership with NBC. But CNET.com is the Web site of interest here (Snap.com was still calling Y2K a virus, seven months after it was over!).

IDG.net

International Data Group (IDG) began in 1964 as International Data Corporation (IDC). *ComputerWorld* (CW) was launched in 1967, quickly unseating *Datamation* as the must-read publication for North American computer types.

Taking the International in their name quite literally, IDG now has 300 magazines and newspapers in 80 countries, mostly local editions of *CW*, *PC World*, *Network World*, and *CIO*. But its biggest growth area has been IDG Books, with its ... *for Dummies* series. In case you haven't noticed, they've spread well past computing topics, to cover such topics as my wife's favourite, *Figure Skating for Dummies*.

IDG.net has taken the same international approach as its magazines, with 270 Web sites in 70 countries. Both the magazines and the Web sites rely on the 24/7 IDG News Service, staffed by 2,000 journalists in 75 countries. And that's not counting the 575 IDC research analysts working in research centres in 43 countries worldwide.

FIRST IMPRESSIONS

ZDNet, at

http://www.zdnet.com

and CNET, at

http://www.cnet.com

both have a familiar look – what many portal reviewers call the Yahoo! look, with major topics in large type with several sub-topics in small type, each a link you can click on like a menu.

IDG.net takes a different approach, focusing on current headlines from the Web sites of IDG magazines, though a few are labelled IDG.net and come directly from the IDG News Service. But the first thing I saw, near the top left corner of the home page, was the option to view the page in a language or region of the world of my choice.

Ironically, the first link I clicked on, *Region*, ran a JavaScript that didn't appear to do anything. The *Language* link changed the list of regions below to a list of languages, and I now found that the *Regions* link restored the list of regions I'd originally seen. But rather than selecting one, I became intrigued by the phrase below the lists:

Map not working? Click here

The flattened globe above turned my cursor into a pointer, instead of a hand with an index finger extended as it does for a link (yes, I have the latest production version of Microsoft Internet Explorer). I clicked on 'Map not working' and was given a choice of four other languages, and about 60 regions. I selected Canada and was greeted by a very different home page from the one I'd been viewing. As you may have guessed, selecting United States takes you back.

On ZDNet's home page, *Product Reviews* is one of the links near the centre of the screen. Further down the page, there's a Yahoo!-like *Reviews* heading that takes you to the same place. Finally, *Reviews* is one of the menu bar items at the top of the page. It may also be tempting to click on *Buying Guides*, but these are from *Computer Shopper*, and focus solely on mainstream hardware.

CNET's home page has two Yahoo!-like headings of interest: *Hardware Reviews* and *Software Reviews*.

With IDG.net, it's a little more difficult. Just to the right of the centre of the screen, you'll see *Top IDG Channels*. Select the one most appropriate for your needs, perhaps *Networking & Systems*. Most channels have sub-channels that allow you to get more topic-specific.

Searching ZDNet

On the ZDNet *Reviews* page is a pair of search boxes near the centre of the screen. The right box defaults to *Reviews*, indicating the search will be made only of product reviews, but you can also select All ZDNet or The Web, though that would not make a lot of sense at this point.

Type in what you want to search for in the left box. Click the GO button, and you get a lot more than reviews listed. In fact, if your screen resolution is 800x600 or less, you won't even see the list of reviews unless you scroll down!

But do scroll down and start in the *Product Reviews* section of the page. If there are more than 12 links, you'll see a *More Product Reviews results...* link at the bottom of the section. It's tempting to click on that link immediately if it's there, thinking that you'll then see a full set of review links all on one page. Unfortunately, however, it doesn't work that way.

When you click on the *More* link, the 7th through 26th entries are displayed, with a link to *Get Results 27-46*. You can scroll, 20 entries at a time, back to the beginning of the year that began three years ago. This year, that means reviews may be dated as early as 1 January 1997.

The links are shown in reverse chronological order, with the newest information first. The title, date, and source are shown, but there's no summary. The advantage of this approach is that 20 entries fit nicely on an 800x600 screen without having to scroll. The disadvantage is that titles don't always tell you enough to determine whether the review is relevant to your needs. Sometimes, the least promising of titles leads to an article with just the information you need.

Although the source is often the name of a ZD magazine, you may also see the word 'products'. In this case, you have to click on the entry to see the name of the magazine displayed just below the title.

In either case, the name of the magazine is useful. If you're familiar with the magazine, you can better gauge how thorough the reviews are. That doesn't mean just sticking to *PC Magazine*, based on my earlier comment about the size of its budget for side-by-side product comparison testing in PC labs. A 'First Looks review' from *PC Magazine* is just what its name implies: a writer spends an hour or three installing, testing, and writing about his experience with a new product. 'Second Looks' are similar.

If you're still only interested in a single source of information, rather than these three broader sources of information, you may prefer to check out these magazines:

- http://www.pcmagazine.com
- http://www.pcworld.com
- http://www.eweek.com (formerly PC Week)
- http://www.computerworld.com
- http://www.networkworld.com
- http://www.macworld.com
- http://www.smartbusinessmagazine.com
- http://www.interactiveweek.com

On *ZDNet Product Reviews*, keep a special eye out for multiple entries from the same date from the same source, especially *PC Magazine*. These are usually the magazine's trademark side-by-side comparisons.

Click on a product of interest and you'll see, not only the review of that product, but a sidebar on the left, broken down into several sections. Highlights typically include:

- *Introduction*. A front page with a one-paragraph summary and links to the rest of the review.
- *Analysis*. Technical overview, and advice on the use of the technology.
- *Review criteria*. The assumptions as to how the technology would typically be used, which drives the selection, testing, and ranking of products.
- *Choosing*. Advice on product selection.
- *Editor's choice*. What they liked best and why.
- *PC Magazine Lab Scorecard*. Rating several aspects of each product, typically as Excellent, Good, or Fair.
- *Summary of features*. Side-by-side comparison of features and capacities, including price.
- *Performance tests*. The results of side-by-side lab tests: numeric values for timing, or throughput values for each product, or N/A if a product failed a particular test.
- *How we tested*. Almost infinite detail on how the tests were performed.

Read Reviews provides direct links to each product's individual review. *Read More* gives a few links to more conceptual information on the technology, competitive technologies, some of the Editors' favourite links on the Internet, and sometimes even a sample RFP (Request for Proposal) for acquiring the technology.

As with virtually all paper magazines that are also available on the Internet, it pays to get your hands on the paper version of a major product review because the Web version may not have all the photos and charts. Because this may require considerable effort and money, it's only really an issue when the decision is a major one. Public and corporate libraries are good places to look. However, if they have microfiche or microfilm, remember that this can make a chart

unreadable by translating two colours into the same shade of grey.

Although it varies by magazine and Web site, things have improved markedly over the last few years, and the newer the article, the less likely it is that you'll need to see the paper copy. But there are some sites that don't give the full version of (some) articles in the misguided view that it will cut into the sales of the magazine. These days, however, it's mostly newspapers that still cling to that view.

Browsing ZDNet

The ZDNet *Reviews* page is well worth bookmarking, at

http://www.zdnet.com/reviews

As well as being able to search from there, you can also navigate the menu-like Yahoo!-like categories.

You may have to scroll to see them, but the *Select a Category* section of the page is highlighted with a background of yellow for easy spotting. There are five categories:

- Computers
- Hardware
- Internet and networking
- Software
- Gadgets.

Don't despair if you can't see the hardware or software category you're looking for. Click *All Hardware* or *All Software* for a complete list.

Although what you'll see next varies substantially, depending on what category you click, here's what you would have seen recently for *Networking Software*. Highlighted in about the centre of the screen is a link to a *PC Magazine* article entitled 'Secure your network'. When you click on it, the centre of the screen is highlighted with a yellow background, listing the first three sections with links for direct access and one-sentence descriptions. A sidebar on the left has links to all seven sections of the article.

Below the highlighted article on the Networking Software page, you'll find a list of products. Don't let the *Check Prices* link for each scare you away – there's a lot more here than links to Egghead and other superstores.

The products are listed with the Editor's Choices first, indicated by red check marks. Click on the product name to see a review. The source of reviews varies, as does the depth. Click on one and you may see a page offering access to full lab test results. Click on another and it will be a brief description from *Computer Shopper*. That can make comparison difficult, but at least you can access them all from the same place.

The Networking Software page also has links to many other useful ZDNet resources, like the Networking Resource Centre, one of a series of IT Resource Centres. The link can be found at the bottom of the right sidebar, under the heading *IT Resources.* In my experience, ZDNet really does make a big effort, with positive results, to help the slightly lost Web browser to regain his/her direction on its site.

Searching CNET

CNET's *Hardware Review* and *Software Review* pages are designed quite differently. From the *Hardware Review* page, you'll find a similar Search function to the one discussed in ZDNet. The right box defaults to *In Hardware*, and you can type in what you want to find in the left box.

Click the Search button and you get a list of products or a 'Sorry! There were no products found that match your search criteria' message. The list of products includes a few technical details, a link to any available review, an estimate of price, and a link to get more detailed price information. For some products, you can immediately check exact prices, while others require that you first configure the product. A box is also provided that allows you to check up to five products for which a side-by-side comparison will be generated.

The *product info* link in the *Review* column will lead you to a page with links to available information. The most popular products are rated

and reviewed, and their performance charted against competitive products. All products have some specs listed: generic information on the type of product, information on the manufacturer, and a facility for users to comment on the product.

In *Software Reviews*, there's a *Best Tech Searching* box in the middle of the screen. Anything typed there will search all of CNET.

But, one level down (for example, if you click on *Networking*), a Search box will appear in the top right corner of the page, in the yellow CNET header. If you change the lower box to *In this category* from its default of *All CNET*, then type your search argument in the top box and click the Go! button, be prepared for a lot. A recent search specifying 'vpn' returned 2,730 pages, each with 25 links on it, many of which were not in English and most of which were not on CNET's site at all, but elsewhere on the Internet.

Browsing CNET

Like Yahoo! and ZDNet, you can click your way down through a menu-like hierarchy of topics and subtopics until you get to where you want to be. Some areas have very few original in-depth reviews by CNET; in some cases, there's little beyond links to other sites on the Internet. Personally, in *Networking* under *Hardware Reviews*, I was happiest with the *All networking reviews* link below the *Options* heading.

Searching IDG.net

On IDG.net, the Search boxes are always the same, and are always present on every page of the site, at the very top of the page, in the centre of the black header. The left box defaults to *All IDG*; the only other choice is a continent, for regional information. Type in your search argument in the right box and click on the Go button.

The 10 most recent matches are displayed, with a count of hits shown near the top left, and an 11-20 link near the top right. For each entry, the title is a link to the full article and is followed by a 2-3 line summary of the article. A relevance percentage value follows, and sometimes

'Editor's Pick'. The date of publication is also shown, though its location varies. About three years of information is stored for searching. A logo to the left of each item shows the magazine where the article was originally published; IDG indicates publication in an IDC research service.

Click on the *Relevance* tab to see the 10 most relevant matches (rather than the most recent). Boxes near the top allow you to specify:

- A subsearch argument, letting you do a second search through just the documents that matched your first search.
- The language(s) of documents returned. The default is English. You can also choose one or more of Spanish, Portuguese, French, and German.

Browsing IDG.net

Once you've navigated to the most relevant sub-channel, you'll usually find recent article titles and short summaries from IDG publications: a mix of customer experience, new product announcements, and product review articles.

SUMMARY

There are, of course, millions of Web sites on the Internet. But each of these three sites brings together a lot of published non-vendor-written product information into one place, and is worth a look, especially for areas where you previously relied on magazines.

A FINAL WORD OF CAUTION

A final word of caution, however: you won't find every product on these sites, so don't expect to use this as an exhaustive search of products that will do a given function. And remember that the farther you stray from mass market hardware and software for individual PC workstation users, the less likely you are to find a complete picture of the hardware or software that's available.

STOP PRESS

As this issue was going to print, CNET announced that it was acquiring ZDNet. Although there will be some consolidation of services, there is a promise that both CNET and ZDNet will continue to exist. Ranking top Internet sites by number of users, they are currently numbers 18 and 19 respectively. Taken together, they will be number 8, with 16.6 million unique monthly users – a 22% share.

Jon E Pearkins
(Canada)

In addition to *TCP/SNA Update*, the Xephon family of *Update* publications now includes *CICS Update*, *MVS Update*, *VSAM Update*, *DB2 Update*, *RACF Update*, *AIX Update*, *Domino Update*, *MQ Update*, *NT Update*, *Oracle Update*, *SQL Server Update*, and *TSO/ISPF Update*. Although the articles published are of a very high standard, the vast majority are not written by professional writers, and we rely heavily on our readers themselves taking the time and trouble to share their experiences with others. Many have discovered that writing an article is not the daunting task that it might appear to be at first glance.

They have found that the effort needed to pass on valuable information to others is more than offset by our generous terms and conditions and the recognition they gain from their fellow professionals. Often, just a few hundred words are sufficient to describe a problem and the steps taken to solve it.

If you have ever experienced any difficulties, or made an interesting discovery, you could receive a cash payment, a free subscription to any of our *Updates*, or a credit against any of Xephon's wide range of products and services, simply by telling us all about it. For a copy of our *Notes for Contributors*, which explains the terms and conditions under which we publish articles, please write to the editor, Fiona Hewitt, at any of the addresses shown on page 2, or e-mail her at fionah@xephon.com

B & M has announced TDSLink, comprising a series of intelligent agents designed to monitor and administer the performance of OS/390 components.

For further information, contact:
B&M Software Europe, Heath End House, West Street, Tadley, Hants, RG26 3ST, UK.
Tel: (0118) 981 1880
URL: http://www.bmeurope.com

* * *

WRQ has announced Version 4.0 of Reflection for the Web, centrally managed Web-to-host terminal emulation packages that Web-enable enterprise applications.

The company has also announced Version 8.0 of Reflection NFS, which enables users to access file and print services on remote systems using TCP/IP with an interface that feels like a PC desktop.

For further information, contact:
WRQ, 1500 Dexter Avenue North, Seattle, WA 98109, USA.
Tel: (206) 217 7100.
WRQ, 40 West Street, Marlow, Bucks, SL7 2NB, UK.
Tel: (01628) 400800.
URL: http://www.wrq.com

* * *

Williams Data Systems has announced the latest version of its RouteView and RouteView Generator SNA subarea and path table generator tools, designed to help move to IP and APPN.

For further information, contact:
William Data Systems, 5 High Street, Old Oxted, Surrey, RH8 9LN, UK.
Tel: (01883) 723 999.
URL:http://www.willdata.com

* * *

IBM has uprated its 3746 multiprotocol controller with increased capacity and improved performance, operator productivity, and network management.

IBM has also announced Version 6.0 of its Communications Server for AIX, with both TCP/IP and SNA enhancements.

It has also announced Version 4.5 of its EDI Services Expedite/CICS for MVS, now with a choice of communicating with Information Exchange using either SNA or TCP/IP protocol.

For further information, contact your local IBM representative, or visit the Web site at http://www.ibm.com/software/network

* * *