



29

TCP/SNA

March 1998

In this issue

- 3 MVS system symbols
 - 8 Networking IBM enterprises with SNA and TCP/IP
 - 21 VTAM LOSTERM parameter
 - 22 The remote systems management challenge
 - 27 A mailbox system for SMTP under MVS TCP/IP
 - 46 Link between NCPs
 - 60 TCP/SNA news
-

© Xephon plc 1998

update

SNA Update

Published by

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

North American office

Xephon
1301 West Highway 407, Suite 201-450
Lewisville, TX 75067, USA
Telephone: 940 455 7050

Australian office

Xephon/RSM
PO Box 6258, Halifax Street
Adelaide, SA 5000
Australia
Telephone: 08 223 1391

Contributions

If you have anything original to say about SNA, or any interesting experience to recount, why not spend an hour or two putting it on paper? The article need not be very long – two or three paragraphs could be sufficient. Not only will you be actively helping the free exchange of information, which benefits all SNA users, but you will also gain professional recognition for your expertise, and the expertise of your colleagues, as well as some material reward in the form of a publication fee – we pay at the rate of £170 (\$250) per 1000 words for all original material published in *SNA Update*. If you would like to know a bit more before starting on an article, write to us at one of the above addresses, and we'll send you full details, without any obligation on your part.

© Xephon plc 1997. All rights reserved. None of the text in this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, without the prior permission of the copyright owner. Subscribers are free to copy any code reproduced in this publication for use in their own installations, but may not sell such code or incorporate it in any commercial product. No part of this publication may be used for any form of advertising, sales promotion, or publicity without the written permission of the publisher. Copying permits are available from Xephon in the form of pressure-sensitive labels, for application to individual copies. A pack of 240 labels costs \$36 (£24), giving a cost per copy of 15 cents (10 pence). To order, contact Xephon at any of the addresses above.

Printed in England.

Editor

Trevor Eddolls

Disclaimer

Readers are cautioned that, although the information in this journal is presented in good faith, neither Xephon nor the organizations or individuals that supplied information in this journal give any warranty or make any representations as to the accuracy of the material it contains. Neither Xephon nor the contributing organizations or individuals accept any liability of any kind howsoever arising out of the use of such material. Readers should satisfy themselves as to the correctness and relevance to their circumstances of all advice, information, code, JCL, EXECs, and other contents of this journal before making any use of it.

Subscriptions and back-issues

A year's subscription to *SNA Update*, comprising four quarterly issues, costs £120.00 in the UK; \$180.00 in the USA and Canada; £124.50 in Europe; £128.00 in Australasia and Japan; and £127.00 elsewhere. In all cases the price includes postage. Individual issues, starting with the March 1991 issue, are available separately to subscribers for £31.00 (\$46.00) each including postage.

SNA Update on-line

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

Welcome to *TCP/SNA Update*

When *SNA Update* was launched, mainframes used VTAM and SNA almost exclusively as the means of communicating with users. With the growth in LANs and the introduction of intranets, this is no longer the case. The majority of sites are looking for people with TCP/IP expertise to hook in these networks to the mainframe. As a result of this broadening of expertise, this is the first issue of the newly-titled *TCP/SNA Update*, which contains information written by and relevant to users of TCP/IP as well as VTAM and SNA.

MVS system symbols

MVS system symbols can be used in VTAM to reduce the number of VTAM definitions that have to be coded in SYS1.VTAMLST. The definitions that can use the symbols are:

- Start option lists
- Configuration lists
- Major nodes
- Minor nodes
- Routing and dynamic reconfiguration definitions
- User definable tables:
 - APPN COS definitions
 - APPN COS to subarea mapping table
 - Associated LU table
 - Message flooding prevention table

- Model name table
- Subarea to APPN COS mapping table.

The text to be substituted is defined in MVS and is put in place of the symbols when VTAM starts or when tables and nodes are activated. This means that by creating the required symbols you can have a single VTAM definition library and members, and use the symbols to control how the system is activated. This should ease maintenance and reduce the number of definitions that you have to update when changes occur. So for multiple systems you can have one set of definitions to maintain. Obviously you must have naming standards that will allow this type of system to be implemented. The symbols can also be used in:

- The TSOKEY00 member of SYS1.PARMLIB.
- VTAM commands, which can assist in operation.
- Sysplex and non-sysplex environments if MVS 5.2 or later is installed.

You can display the symbols that are available on a system by issuing the MVS DISPLAY SYMBOLS command. Additional symbols can be defined as required. The symbols are best utilized for defining system identifiers and names when coding:

- Start options
- Configuration lists
- Major and minor node definitions
- Table definitions.

The MVS static symbols that are provided as part of the system are:

- &SYSCONE – shorthand system name (1-2 characters). Defined in IEASYM_{xx} in SYS1.PARMLIB (defaults to the last two characters of &SYSNAME).
- &SYSNAME – the name of the system (1-8 characters). Defined in IEASYM_{xx} or IEASYS_{xx} in SYS1.PARMLIB (defaults to the processor identifier).

- `&SYSPLEX` – the sysplex name (1-8 characters). Defined in the `COUPLExx` or `LOADxx` member of `SYS1.PARMLIB`. If `LOADxx` does not specify the name, it defaults to `LOCAL` until `COUPLExx` is processed.
- `&userdefined` – maximum of 100 symbols can be defined (1-8 characters). Defined in `IEASYMxx` member of `SYS1.PARMLIB`.

The display symbols command and its output look like this:

```
DISPLAY SYMBOLS
```

```
IEA007I STATIC SYSTEM SYMBOL VALUES 483
      &SYSCONE.      =  "00"
      &SYSNAME.     =  "SYS0"
      &SYSPLEX.    =  "0PLEX"
```

The *MVS Initialization and Tuning Reference* details how and where to code MVS system symbols. The symbols must always begin with an ampersand (&) and end with a full stop or period (.). The maximum number of characters that can appear between the ampersand and the period is 1-8. After MVS substitutes the symbols with the text, VTAM will be responsible for ensuring they meet VTAM naming requirements. This means that, if the symbols are too long or coded after existing values, errors could occur. However, they would not be reported when the symbol is coded, but when VTAM tries to action them after substitution has taken place. For example, if you code `&SYSNAME.` as 01 and then code `CONFIG=01&SYSNAME.`, an error would occur when VTAM is started because `CONFIG=0101` would be the result of the substitution – and this is invalid.

To show how the symbols can ease the definition problem, consider the following systems. Two MVS systems exist in a sysplex environment. Each host will have a separate VTAM and run the same application programs on each system. This would require the following definitions to be coded:

- Host number one:
 - `ATCSTR01`

This would contain:

```
CONFIG=01,  
SSCPID=01,  
SSCPNAME=HOST01
```

- ATCCON01

This would contain APPLS01 and other major nodes.

- APPLS01

This would contain a list of APPLs, for example:

```
APPL01A  
APPL01B
```

- Host number two:

- ATCSTR02

This would contain:

```
CONFIG=02,  
SSCPID=02,  
SSCPNAME=HOST02
```

- ATCCON02

This would contain APPLS02 and other major nodes.

- APPLS02

This would contain a list of APPLs, for example:

```
APPL02A  
APPL02B
```

When VTAM is started, the operator on host one would issue 'S ACFNET,LIST=01', and for host two they would issue 'S ACFNET,LIST=02'.

This means each system has a start options list, a configuration list, and separate major nodes for the application programs, and different start commands have to be remembered. If we utilize MVS system symbols, it is possible to have one set of definitions for each system and one command to start the system.

So you could have a single ATCSTR00 and this would contain:

```
CONFIG=00,  
SSCPID=&SYSCLONE.,  
SSCPNAME=&SYSNAME.
```

The single ATCCON00 member could have a major node named APPLS and this major node could have the names of the APPLs coded as:

```
APPL&SYSCLONE.A  
APPL&SYSCLONE.B
```

The operator then would only have to issue 'S ACFNET' on the system in question. The default is to read ATCSTR00.

The symbols would then resolve as follows for host one:

```
SSCPID=01,  
SSCPNAME=HOST01
```

```
APPL01A  
APPL01B
```

On host two they would resolve as:

```
SSCPID=02,  
SSCPNAME=02
```

```
APPL02A  
APPL02B
```

As a result of using some simple coding techniques you can reduce the VTAM maintenance and reduce the procedures required by operators for different systems. You can also reduce errors should operators become confused as to which system they are working on – which is easy to do if the operation being performed is during unsociable hours.

By defining user symbols you can enhance the system further, probably reducing many major nodes and enabling one definition library to be utilized rather than many across systems.

John Bradley
Systems Programmer (UK)

© Xephon 1998

Networking IBM enterprises with SNA and TCP/IP

INTRODUCTION

Only a few years ago many analysts and industry watchers predicted a premature death for SNA. However, based on the recent statistics, these estimates seem to be a bit over-exaggerated.

Although the number of TCP/IP installations is growing at an exponential rate (almost all PC-based communications packages include an IP stack), not too many of the 50,000+ SNA installations completely migrated their mission-critical applications to IP. On the contrary, a recent IDC survey shows that the number of SNA gateway installations continues to grow at a 15–20 percent rate and 68 percent of networking budget is spent by SNA managers. So for the next few years, the network infrastructure for most organizations will include a combination of SNA/APPN and TCP/IP.

Since few companies have the luxury of turning off their existing network infrastructure and waiting for the emerging new single networking architecture, the real question for network managers today is: what is the most effective way of supporting new applications over the existing infrastructure? Fortunately for the users, there are a handful of good answers.

HISTORICAL OVERVIEW

Following a successful deployment of new applications on the LANs, there was a need to connect them over the WANs. Shortly after, the developers and users realized that, unfortunately, the bandwidth of the WAN was not sufficient for LAN-based protocols. Most of the LAN protocols (AppleTalk, NetBIOS, and IPX) were too chatty and could cause so-called broadcast storms. The next natural step was to utilize the least expensive and most available protocol for WAN communications. However, there was no magic relief with TCP/IP either. The overhead was significantly higher than it was with SNA. While this was not a real problem on LANs, it became a major headache on the WANs. An additional problem is with TCP/IP's use

of congestion detection as opposed to the congestion avoidance used in SNA; TCP/IP networks are more susceptible to runaway overloads at high line utilization. Other problems are introduced by the ancient, but still the most popular, IP Routing Information Protocol (RIP). As a distance-vector protocol, RIP is inflexible; it introduces additional overhead; and it can create loops and black holes in complex IP networks.

The traditional (pre-APPN) SNA, commonly associated with 3270 applications, is a connection-oriented protocol with routing only at the fourth (ie transport) layer of the OSI model (equivalent to the transmission control layer in SNA). The downside is that this layer lacks dynamic re-routing capabilities, and high intelligence is required in all routing nodes. APPN today is usually associated with application-to-application (peer-to-peer) networking. It employs a connection-oriented Intermediate Session Routing (ISR) link state protocol. Link state routers (network nodes) are characterized by a topology database, containing information about states of all network links included in each network node. Each node is responsible for computing the best routes to any other node in the network. APPN network nodes are responsible for maintaining the topology database, integrity of which is preserved by using sessions between adjacent Network Nodes (NN). This is different from distance-vector-based protocols (eg RIP), where such information is periodically broadcast (every thirty seconds with RIP), regardless of whether there were any changes. The more advanced TCP/IP link-state-based routing protocols (such as Open Shortest Path First [OSPF]) also broadcast their routing information when a link state is changed. However, they also periodically broadcast this information (every thirty minutes) regardless of changes in the link states.

When a session is established between two nodes in the APPN network, it will live as long as the preferred route (selected during the session set-up) is available. The major drawback of this method is that the preferred route, during a long-lived session set-up, will not necessarily remain the optimal route for the life of the session. There is also no re-routing around failures (ie sessions have to be restarted even because of an intermediary node failure), which may potentially disrupt a client/server application.

To address the major concerns of SNA/APPN, in May of 1995, IBM came out with their enhancement for APPN, HPR. HPR is a connectionless protocol, allowing packets to be dropped during network congestion and non-disruptive re-routing around failures. Rapid Transport Protocol (RTP) is used at the end-points of an APPN HPR session to guarantee delivery and resequencing of packets. An Adaptive Rate Based (ARB) flow control mechanism is used at the end points of an HPR session to prevent network congestion. HPR can be mixed with ISR nodes within an SNA/APPN network.

WHAT'S HAPPENING ON THE LAN?

What's happening on the LAN? Firstly, it's migration from shared to switched media. At the same time we can see a trend for migration to higher bandwidth: 10/100/1000MB Ethernet, Token Ring-in-Fast Ethernet, and ATM. With the migration from shared to switched LANs, the term VLAN has become a common term – not only within standards committees and engineering departments, but in many network management centres and campuses. For quite some time Virtual LANs (VLANs) were considered 'marketecture', invented by vendors of switched equipment and, because of their inherent complexity, very rarely implemented by end users. Today, however, as switched LAN infrastructures slowly but surely replace shared LANs, even industry analysts have discovered that VLANs are required for the very reasons vendors were promoting them – to enhance security and traffic control; to ease network adds, moves, and changes; to contain broadcasts; or, summarizing all of the above, to enhance manageability of switched LANs.

IN SEARCH OF SOLUTIONS

Since the primary objective of implementing VLANs is to enhance network manageability during the network planning and design stages, centralized VLAN management is an important requirement. When VLANs can be defined remotely and managed from a central location, network managers can more easily design their networks based on business objectives, such as improved service to users, while

also continuously monitoring VLAN performance and adjusting VLAN policies and definitions.

Such tools should be capable of centralized network administration, including the ability to control and define initial assignments of users per VLAN, as well as the ability to manage on-going adds, moves, and changes.

Dependable VLAN management tools can substantially reduce or even eliminate the need for manual administration of users' adds, moves, and changes. For example, when a user moves within the network, the switches will sniff a user's MAC address and automatically assign the user to the correct VLAN. Additionally, VLANs can significantly improve security management by automatically placing unrecognized network users into a default VLAN, with minimal accessibility, secure from the rest of the network. For example, when any user plugs into any switch port, VLAN management software on the switch will check the user's MAC address against a predefined VLAN map; and if the user's MAC address is unrecognized, the switch will both alert the network management station and place the user in a default VLAN for insecure users.

The same functionality also resolves a problem for unrecognized mobile users, who will not be able to bypass security checks by connecting into a secure port. Such exposure exists when VLAN membership is solely determined by a switch port. Efficient VLAN management can improve productivity and highly simplify procedures for mobile users, ensuring optimal service and performance, no matter what switch port they log into.

STANDARDS CAN HELP SOLVE THE INTEROPERABILITY CHALLENGE

Currently there are several VLAN management products on the market. However, one of the problems faced by end users is the fact that these products usually do not interoperate, and only support a specific vendor's equipment. To address this issue IEEE (<http://www.ieee.org>) is formalizing a new standard, currently in draft status, for Virtual Bridged LANs entitled IEEE 802.1Q. Although it's far from an ideal or complete solution, this standard is a first step towards

introducing multi-vendor standardization for VLANs. IEEE 802.1Q does address the important issue of VLAN tagging. Generally speaking, there are two ways to tag frames:

- Implicit Tagging – a frame is classified as belonging to a particular VLAN based on the data content of the frame (eg MAC address, layer 3 protocol) and/or the receiving port.
- Explicit Tagging – a frame is classified as belonging to a particular VLAN based on an explicit VLAN tag value that is included in the frame.

Even though IEEE 802.1Q currently concerns itself with Explicit Tagging, it also provides a means of implicit frame classification based on the receiving ports. To address the issues of multi-vendor support, and different levels of smartness contained in vendor equipment, the forthcoming standard will have to address the issues of interoperability between relatively simple and sophisticated high-function switches. For example, switches that only understand port-based VLANs are forced to explicitly tag the frames based on the port of entry, even if the frame may already have been tagged by the end station or another switch. On the other hand, high-function switches may look at the subnet mask of the IP packets in IP subnet-based VLANs and decide which VLAN the traffic belongs to, based on that information. These switches may then decide whether or not to explicitly tag the frames, since both tagged and untagged frames can be forwarded, although not on the same VLAN. The rationale behind allowing switches the freedom to apply their own level of VLAN membership awareness is the fact that, in a multi-vendor environment, progress rules must remain most flexible to provide painless frame forwarding.

The forthcoming IEEE 802.1Q standard is compatible with the transparent bridging standard, IEEE 802.1d. Therefore, the new standard specifies that legacy bridges, unaware of VLANs, can be incorporated into networks with VLAN-aware devices. For example, when a legacy bridge is placed between two tag-aware devices, its job is to forward the traffic as usual, without interpreting the tag value. According to IEEE 802.1d, a legacy transparent bridge will flood the unknown destination MAC addresses to all ports.

IEEE 802.1Q is expected to be finalized in the middle of 1998, and the final draft document should be available quite soon, allowing vendors to finalize their strategies on the migration/integration of the emerging VLAN tagging standard with their existing products.

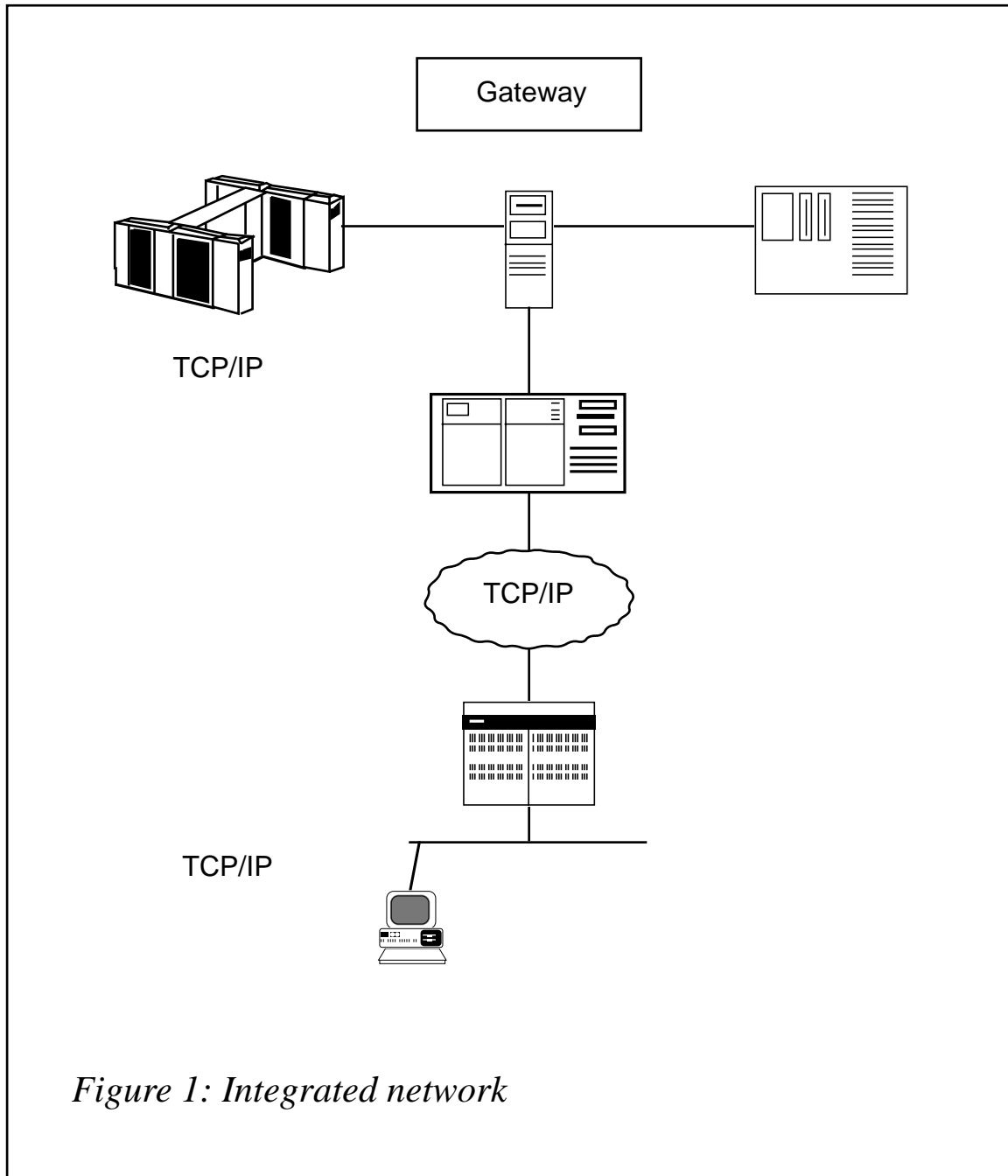
VLANs are necessary for intelligent migration from shared to switched LAN infrastructure. They introduce a significant step towards the concept of self-managed networks. Not only can VLANs be defined and used, more importantly, VLANs can be reliably managed, here and now by the existing and emerging network management tools. When selecting a reliable vendor for VLAN equipment and management platform, it's important not only to learn the functionality and richness of existing tools, but also to study the vendor's migration/integration plans for these tools in conjunction with the emerging IEEE 802.1Q standard.

NETWORK CONSOLIDATION APPROACHES

The number of TCP/IP-based client/server applications and LAN-based applications is constantly growing; in most cases one can find a mix of SNA and non-SNA (ie TCP/IP, IPX/SPX, NetBIOS, Banyan VINES, AppleTalk, etc) networks next to each other. Such a network strategy is probably safe but too costly. Rather than maintaining parallel networks for different protocols (that, at the extreme, can cost twice as much as a single network), networks can now be combined without impacting application support – see Figure 1.

Many vendors realized the importance of consolidation of the network infrastructure and the potential savings, and developed products allowing utilization of non-SNA networks for SNA traffic. The most popular approaches for multiprotocol network consolidation include SNA transport over IP with Data Link Switching (DLSw), Frame Relay with Request for Comments (RFC) 1490, and LAN transport over SNA/APPN backbone.

Multiprotocol routers with DLSw are some of the more important examples of the SNA-over-IP approach, which was originally introduced by IBM at the beginning of 1993, and in March of the same year adopted by the Internet Engineering Task Force (IETF) as RFC



1434. This RFC for DLSw was enhanced and superseded by RFC 1795 in April 1995. Because of the high timing sensitivity of SNA sessions, DLSw is the only dependable technique for transporting SNA over IP networks. Actually, encapsulating SNA data into IP frames is not new to the router vendors, but similar pre-DLSw attempts ran into time-out problems. IBM created DLSw to address this shortcoming, reducing the likelihood of time-outs by providing a

local polling and logical local termination of the data link (often called spoofing). By keeping the acknowledgment local, routers also reduce the amount of traffic traversing the wide area link. Additional advantages of DLSw include the ability to route around link failure, support for an extended source route bridging hop count, and NetBIOS name cacheing, which greatly reduces NetBIOS broadcasts.

To address scalability and improve connection set-up time, AIW (APPN Implementors Workshop) developed DLSw Version 2 – accepted by IETF as RFC 2166 in July of 1997. Among the major benefits of DLSw V2 are reduced bandwidth requirements for set-up and connections, improved memory utilization, reduction in TCP overhead and number of tunnels, and reduced overhead for NetBIOS. 3Com authored the RFC and implements DLSw V2 on its NetBuilder family of products.

SNA GATEWAY SOLUTIONS

As enterprise networks have evolved over the years, different technologies, such as departmental LANs, have been added to the SNA backbone, creating integration challenges. One of the key problems in today's environment is providing access for PC users on LANs to legacy SNA hosts, requiring 3270/5250 sessions. One of the most popular solutions to this problem is the deployment of SNA gateways. Today, however, many gateways are not just limited to providing host access, but also provide additional value-added functions. Bundled applications, FTP servers, distributed transaction processing, and Web-to-host connectivity are among the more interesting value-added functions.

An innovative solution, providing customers with high-speed LAN connectivity to SNA and TCP/IP mainframes, was recently jointly announced by 3Com and Bus-Tech. This new solution is supported by the major vendors of SNA gateway software – IBM and Microsoft. Higher-speed communication is needed in today's networks. Gigabit Ethernet is evolving to become the most prevalent LAN backbone technology in the industry. In addition, given that Token Ring technology is limited to 16Mbps, TIF provides a method of transporting

Token Ring in frames across a 100Mbps Ethernet link, thereby providing Token Ring customers with a migration path to higher-speed bandwidth. Another benefit of TIF is its ability to make migration from Token Ring to Ethernet transparent to the network. The synergy of the new enterprise connectivity solution this relationship brings to the customers is based on the expertise each one of the companies has demonstrated in their respective areas. Integrating Bus-Tech's NetShuttle and Microsoft SNA Server with TranscendWare will provide enterprises with an easy-to-manage, highly reliable, and cost-effective method of LAN-to-host and intranet-to-mainframe integration. This connectivity also addresses customer needs for end-to-end network management. Future plans are being developed to integrate the management of channel connections by 3Com's TranscendWare Enterprise Manager and to further integrate Microsoft BackOffice applications with SNA Server. Bus-Tech's NetShuttle family of plug-and-play intranet-to-mainframe connectivity products will be the platform by which these capabilities are integrated.

To allow easier integration between LAN and mainframe applications, users will be able to utilize interfaces between host mainframe applications, databases, files, and security systems with Microsoft BackOffice suite of products: SQL Server for database access, Proxy Server for security, Transaction Server for application infrastructure, and Site Server for deploying and managing Intranet sites.

EMERGING TECHNOLOGIES

Over the years, many IT organizations tried to extend access to their established host systems and applications. For example, during the mid to late 1980s ASCII terminal emulation software and dial-up 3270 protocol conversion products became common in SNA environments. Although this approach was saving money in hardware expenditure, most of these efforts achieved only limited, or short-lived, success. To a large extent, the cost-benefits of the less expensive equipment was surpassed by the extensive cost of end-user training, maintenance, and support issues. For example, training the end users how to navigate and interact with a host application by using a sequence of ASCII keystrokes that simulated certain mainframe

terminal keystrokes turned out to be far more difficult than expected. Thus, many enterprises decided that the cost of supporting such a capability outweighed the benefits (at least for large-scale use). Today's emerging Web access technologies solve these problems. Therefore, some of the recent developments in integration of SNA and the multiprotocol network world include distributed network management, distributed presentation with Java, and Web integration. The Internet's incredible popularity was sparked by the widespread availability of Web browsers. The Internet has been an incredibly rich source of information for scientists and engineers for quite a few years, but the interfaces to access this information (ie Telnet, FTP, Gopher, etc) were not too user-friendly. Only when Web browsers became common, did the Internet become so popular, and, along with this growing popularity of the Internet, we are finding new implementations for the Internet and Web browsers almost every day. One such new implementation, becoming more and more popular, is Web-to-host integration.

One of the most substantial advantages of Web-to-host integration is improved manageability and reduced development time for the new applications by enabling the use of advanced and compelling Web administration and development tools. For example, programming for distributed presentation, distributed transactions, and distributed database access, which represents quite a challenge in the legacy world, can be simplified by leveraging currently available programming skills and tools, such as HTML, Java, ActiveX, etc.

Different models and applications for Web-to-host integration can be identified today:

- Publishing – dissemination of relatively static information: company policies and procedures, standards, manuals, and so on.
- Database integration – access and integration to relational and object-oriented data residing on the host for inquiries and updates from standard user workstations.
- Application access – ability to provide immediate access to host applications with minimal changes to the host and host-based applications. Among the challenges to be resolved during

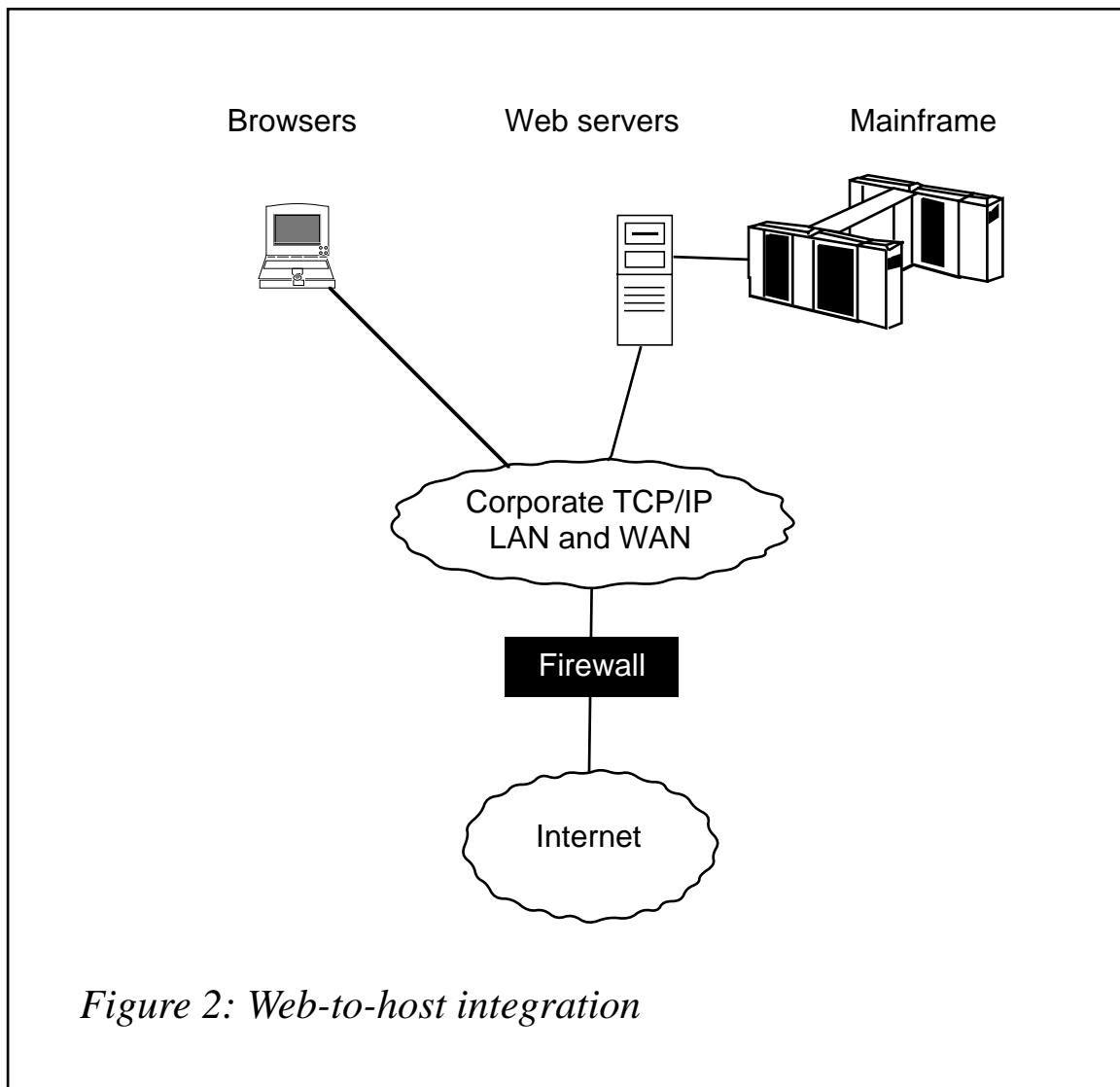


Figure 2: Web-to-host integration

implementation is mapping of keystrokes and synchronization with browser buttons, as well as addressing security issues by incorporating secure sockets layers (SSL) and secure HTTP (S-HTTP) standards. This is illustrated in Figure 2.

Some specific suggestions on how to proceed with Web-to-host integration:

- Document your current environment.
- Develop a current-to-future gap analysis; quantify costs and benefits of migration.
- Refine and compare alternative deployment models and architectures.

- Consolidate security schemes.
- Evaluate Web application design tools; select the appropriate one.
- Design, implement, and test a pilot Web-to-host integration program.

Web-to-host integration tools will also enable advanced functionality, such as form validation, re-engineered workflow, integration of multiple hosts, and broader applications (server and/or desktop-based multimedia applications, for example) into a single screen. That results in improved productivity, reduced training and support requirements, and even saving some trees (viewing output on the desktop reduces printing requirements). Additional savings can be achieved because Web development costs less than the traditional mainframe and client/server development.

The bottom line is that this modernization supports a more efficient movement of information both inside and outside the company.

SUMMARY

As shown in this article, merging dissimilar technologies of classic SNA, the new APPN/HPR, and TCP/IP is a very hot item on the agendas of enterprise network managers in many organizations today. However, we have also demonstrated some of the most popular solutions for network integration, available here and now! One of the conventional approaches taken by network managers is maintaining dual network infrastructures (with excess capacity for non-SNA links) – one for deterministic, bandwidth-efficient, mission-critical SNA applications and another one for non-deterministic, bandwidth-hungry LAN and IP networks. Because of their worries about congestion, the main cause of data loss and unpredictable performance, most network managers play it safe. However, with products that efficiently utilize SNA/APPN architecture, SNA-centric networks will not succumb to congestion, and the costly solution of ordering excess capacity for the links can be avoided. Such products are especially attractive for users running a traditional SNA or SNA/

APPN WAN backbone. Although many already have a multiprotocol network, it is expected that most of them will still have SNA/APPN as the predominant network architecture for mission-critical applications in the foreseeable future. By providing a multiprotocol solution based on SNA/APPN or TCP/IP with DLSw, the requirement for transporting multiple protocols across a WAN can be satisfied. And there is a certain trend for integration of segregated multiprotocol networking infrastructures into a single multiprotocol network, with TCP/IP playing an even more important role for supporting mission-critical data.

Indubitably, the Internet presents an important new approach to boosting the value of data and applications residing in the 'glass house', by making them more widely available, easily utilized, and accessible via the Internet. The key to successful Web-to-host integration lies in thorough implementation planning and careful selection among the different implementation alternatives, to better suit a multitude of company-specific integration realities and needs. Given the recent developments of Web-to-host solutions, incorporating legacy systems into a company's Internet and/or intranet strategy is not only viable but also a cost-effective approach. The Internet has presented not just a new computing paradigm, but also a way to combine legacy data with new presentation techniques, thereby leveraging an enormous information repository with a new network of both internal and external users. These new developments allow cost-effective and user-friendly integration of SNA with new Internet and Web-based technologies. Such developments assist the users to design and implement reliable and efficient networking solutions, making use of the SNA/APPN knowledge base and providing a dependable migration path to future technologies.

REFERENCES

SNA and TCP/IP Enterprise Networking, eds. Daniel C Lynch, James P Gray, and Edward Rabinovitch, Manning Publications, ISBN 1884777112, 1997.

John Poston
Network Consultant
3Com (UK)

© 3Com 1998

VTAM LOSTERM parameter

In the September 1997 issue of *SNA Update* in an article entitled *VTAM LOSTERM start parameter*, I incorrectly wrote that the VTAM system incorporated a new start-up parameter named LOSTERM. This parameter is actually a part of the application major node definition.

The actual content of the article and how the parameter works is correct, but it is not a start-up option. The following details again how the LOSTERM= parameter can be coded.

An application that has a LOSTERM exit routine but no NSEXIT routine must issue a CLSDST macro instruction to terminate a lost session. Some applications do not do this and this results in session hangs. Even a VTAM VARY INACT command will only result in the logical unit being put into a PENDING NOTIFY STATE. The only solution in the past has been to recycle the application or to restart VTAM – both being rather heavy-handed to resolve an issue for one logical unit.

The new LOSTERM= option can be coded on the VTAMAPPL major node statement and has three options. These are:

- LOSTERM=NORMAL
- LOSTERM=IMMED
- LOSTERM=SECOND.

LOSTERM=NORMAL means that VTAM will perform the termination.

LOSTERM=IMMED means that VTAM will send an UNBIND to break the session on the first call to the LOSTERM exit.

LOSTERM=SECOND means that VTAM allows the LOSTERM exit to issue a CLSDST macro instruction on the first call, and if a second call is initiated then will send an UNBIND to terminate the session.

John Bradley
Systems Programmer (UK)

© Xephon 1998

The remote systems management challenge

For years, pioneering companies have been extending corporate information systems to reach out to remote sites and mobile workers, and provide access to customers, suppliers, and partners over the Internet. The trend is confirmed by the huge growth in laptop computing, remote access connections, and the use of GSM phones for data. These extended organizations are destined to become the norm, rather than the exception.

Although, according to research conducted by the Gartner Group, less than 20% of white-collar workers within today's organizations are 'location independent', this figure is set to more than double to 43% by the turn of the century. Technological changes, together with business and social pressures, are combining to create an 'extended workplace' in which as many as three-quarters of senior professional managers and knowledge workers will be mobile.

Waves of remote computing, often starting with remote sites, and extending to mobile workers, World Wide Web access, and corporate intranets and extranets, are washing over many organizations. It is a trend that can only accelerate – and it is the administrators responsible for the management and control of corporate IT systems who will most keenly feel the impact.

Within five years, more than 100 million workers around the world will regularly work outside the boundaries of the enterprise, without the comfort of continuous LAN or high-speed WAN connections. The workplace will change forever, and this army of 'location independent' workers will share one common characteristic: they will be only intermittently connected to the corporate information systems they depend upon to do their job.

For the IT department, this revolution will mean that the percentage of budget allocated to supporting bandwidth-challenged remote and mobile users will grow from an average of less than 5% in 1996 to more than 30% by the turn of the century. The management of remote and mobile computing will move from a tactical response to a

strategic imperative for many organizations – and unfortunately, the systems management issues involved will not be solved simply by extending the LAN- and WAN-centric thinking that pervades most enterprise systems management architectures today.

For those responsible for supporting remote and mobile users within the ‘extended enterprise’, a handful of key factors define the difference in their working environment.

LAN-based computer users typically benefit from unlimited, continuous access to information, applications, and systems. They rarely have to worry about synchronizing client and server data, are locally managed and supported, and have come to expect fast, cheap, and reliable network connections.

Compare their lot to that of the remote or mobile worker; they have to deal with limited intermittent access to corporate systems, suffer unpredictable connection windows, and need to do much of their work off-line. They have to worry about how and when client and server data will be synchronized, generate complex management and support problems, and labour under slow, unreliable, and expensive connections.

Over a typical dial-up line at 28Kbs, connection speeds are more than 300 times slower than the minimum of 10Mbps that is normal for a local area network. Add to that the fact that mobile users simply don’t want (and can’t afford) to be on-line all the time, and the throughput gap widens to as much as 1:10,000.

No-one should imagine that these order-of-magnitude bandwidth limitations are going to disappear any time soon. According to Gartner Group and others, well into the next century the majority of remote and mobile users will still be connected to the enterprise at speeds that are hundreds of times slower than a LAN connection. They will have to remain disconnected most of the time, and they will not have the luxury of calling on IS help from ‘down the corridor’ if they have a problem. And unlike users on the LAN, where the bandwidth is effectively ‘free’ once the system has been installed, remote and mobile users will be paying for every second of connect time they use.

For most IT professionals, managing dial-up systems over a slow remote access connection and using client/server software that was never designed to work at less than LAN speeds is going to be an increasingly painful experience, and it isn't going to be a lot of fun for the user, either.

They will have to wrestle with problems like:

- How can we guarantee that the latest version of our flagship corporate application is installed in all 1,200 remote locations overnight?
- Key customer data has been corrupted on a salesman's laptop: how can we restore it in time for a multi-million pound sales presentation in less than two hours' time?
- A deadly new virus has been detected: how can we ensure that the signature is delivered to all remote users in time to intercept it?
- We're about to install Office 97. How much disk space remains on our fleet of salesforce laptops – all 500 of them?

It is no surprise that Gartner Group estimates that the five-year cost-of-ownership of typical mobile users will be as much as 58% higher than for their LAN-based equivalent. The difference is driven by significantly higher capital, support, administration, and end-user costs.

It is clear that the management and support of these remote and mobile workers requires a new architectural approach – an approach that focuses as much on keeping them productive while they are off-line as it does on providing an efficient connection when they are on-line. Systems administrators want to be able to effectively manage remote systems and applications, reduce cost-of-ownership, and maximize end-user productivity.

The restoration of management control requires that the classic elements of systems management – software distribution, asset and configuration management, content and applications management, diagnostics and recovery, and event and alarm management – be extended to all remote and mobile systems. It must be done in a way

that is optimized for users who connect intermittently over a slow-speed link.

Conventional client/server models can't achieve this. Client-agent-server, queued event architectures provide the necessary optimized foundation. The client-agent-server approach incorporates agents that automatically queue and stage system events. It allows remote users to work autonomously while off-line, and minimizes time and bandwidth requirements once a connection has been established.

Associated session and communications management services allow system administrators to establish priority, cost, and security considerations for all transactions and data types. The connection method and network protocol used can be matched to each requirement. During the session, intelligent 'work objects' define communications windows and retry rules, determine pre-, during-, and post-session processing rules, update client files and registries, and execute programs. Automatic data compression, checkpoint restart, and incremental updates make the best use of limited bandwidth and unreliable connections.

Actionable management information is key to effective management control, and the administrators of remote systems require particularly comprehensive session logging and reporting capabilities. All actions must be automatically date-and-time stamped, and a detailed audit trail must be provided for all actions. By highlighting incomplete actions and error conditions, system administrators must be equipped to 'manage by exception'.

So how does client-agent-server based remote systems management work in action? While systems are off-line, it runs quietly in the background, allowing the system manager to queue up software upgrades, updated Web content, new price files, and new anti-virus scripts. It can automatically monitor for alarm conditions, and prepare to repair corrupt or damaged files. At the same time, the off-line client system can be checking for free disk space, installing software, requesting a new Web page, reporting a virus attack, staging an incremental back-up, queuing an e-mail, or capturing a sales order.

As soon as the remote site or mobile user has connected – regardless of the communication method employed – data is automatically compressed and synchronized between client and server in a highly efficient manner. The use of checkpoint restart ensures that, even if the line is disconnected during the call (not an uncommon experience for GSM users!), the connection is automatically restored to the precise point it had reached – there is no need to have to start the process all over again. Once the connection has been completed, changes are automatically applied and the system administrator provided with a comprehensive set of management information and alarms that allows them to pinpoint and deal with any failures and error conditions.

Remote systems management should not be restricted just to managing specific groups of remote and mobile users – it must also embrace leading enterprise systems management solutions, and extend their capabilities to remote and mobile users throughout the organization. It requires active partnership with leading hardware, remote access, and application vendors to improve the manageability of their solutions in a remote or mobile environment.

Client-agent-server systems management architectures are optimized for intermittent, low-bandwidth connections. They simplify the management of remote sites and users, significantly reduce capital and communications costs, and improve user and administrator productivity. Both the user and the administrator see real benefits. Studies suggest that an effective remote and mobile systems management regime can allow salespeople to divert time spent from administrative tasks into face-to-face selling, such that productivity improvements of 25-100% can be achieved. Equally, systems administrators are equipped to support two to three times more mobile users in such a management environment. And communications time and cost savings of 80% or more have been achieved.

Remote and mobile computing is fast becoming a way of life. For an increasing number of organizations and workers, the data is out there – somewhere!

Bob Apollo
Vice-President and Managing Director
XcelleNet International (USA)

© XcelleNet International 1998

A mailbox system for SMTP under MVS TCP/IP

This article describes an implementation of a mailbox system for SMTP, based on ISPF functions.

Basic MVS TCP/IP SMTP (Simple Mail Transfer Protocol), as it comes from IBM, does not provide any form of mailbox system. SMTP comes as an option only under TCP/IP and runs as a started task.

The only standard user-interface to SMTP that is provided is the REXX-list SMTPNOTE, which enables mail to be edited using correct SMTP-syntax, and then calls the TSO command TRANSMIT to submit the mail to SMTP. This REXX list is based on the old TSO line mode editor and is very basic and primitive in today's world of graphic interfaces.

The receive function is solely based on the TSO command RECEIVE, which is primarily a line mode command only.

The mailbox system is implemented as a PDS or, as I would recommend, as a PDSE (with all the advantages of PDSEs), where all mail, transmitted as well as received, is stored with appropriate ISPF statistics. The ID field (user-id) of the statistics is used to indicate which type of mail it is, and the names of the members are derived from the name in the subject field of the mail.

The mailbox system is accessed on-line in TSO as an ISPF selection, preferably as an ISPF application because it uses special PF keys in certain functions. Alternatively, a special keylist can be set up. Access to mail is through the standard ISPF editor.

The main functions in the mailbox system are to:

- Receive and view incoming mail from SMTP.
- Build new mail according to RFC standards and include mail text and submit to SMTP for transmission to the receiver(s).
- Reply to incoming mail.
- Forward read mail to other receivers.

- Delegate incoming mail to a stand-in mailbox.
- Search in a mailbox directory.

Even though the mailbox system is primarily an on-line system, a batch interface is also supported. The two basic functions, construct/send mail and receive mail, are supported in batch. Send is supported under TSO or ISPF in batch via the CLIST MAILSENS, while receive is supported under ISPF in batch via the CLIST MAILRECV. These CLISTs are used in the same way under TSO on-line for send and receive functions.

The mailbox system was developed and executes on an MVS system located in Scandinavia, which has special national language support, ie there are three extra letters to support in lower and upper case.

The SMTP system communicates with an in-house cc:Mail system in Scandinavia as well as an in-house cc:Mail system in the USA. Because it is also set up for general e-mail access to and from the Internet, it was decided to use only the English alphabet in mail addresses and mail text. Therefore, a standard US-English EBCDIC/ASCII translation is selected, without any support for national characters.

The most important thing to understand here is the basic formal syntax of mail; the details are described in the TCP/IP protocol descriptions RFC 821 and 822.

The normal contents of mail includes receiver, which is optional and only required if cc-receiver(s) is requested. Other types of fields, as described in the RFCs, can exist. This is illustrated below:

```

helo sender-domain
mail from:<sender-userid@sender-domain>
rcpt to:<receiver@receiver-domain>
rcpt to:<cc-receiver@cc-receiver-domain>
data
Date: DD month YYYY HH:MM:SS GMT-offset/time-zone
From: sender-userid@sender-domain
To: receiver@receiver-domain
Cc: cc-receiver@cc-receiver-domain
Subject: name of mail

```

text
text
text

To submit mail to SMTP on-line or in batch build the mail in a dataset and issue the following TSO command:

```
TRANSMIT smtpnode.stcname +  
  DATASET('mail-dataset-name') NOEPILOG NOLOG NOPROLOG
```

where: 'SMTPNODE' is the name of the JES node where SMTP is running as a started task (it can be on a different node if running in an NJE-network); 'stcname' is the name of the started task running SMTP; and 'mail-dataset-name' is the dataset containing the mail to be sent.

In batch, mail can also be sent via JCL and a similar method can be used in TSO using appropriate allocation statements:

```
//S1      EXEC PGM=IEBGENER  
//SYSPRINT DD  SYSOUT=*  
//SYSIN   DD  DUMMY  
//SYSUT2  DD  SYSOUT=B,DEST=stcname of SMTP  
//SYSUT1  DD  DATA,DLM=ZZ  
mail-header  
mail-text  
ZZ
```

To receive mail on-line or in batch use the RECEIVE command:

```
RECEIVE USERID(addressee-userid)
```

To receive on behalf of another user, ie using the parameter USERID(addressee-userid), the user-id of the receiver must have RACF authority to use the TSO OPER command.

Incoming mail is placed on the spool in output class B (or any other designated punch class if so defined in the SMTP set-up parameters) with jobname equal to the name of the started task running SMTP and destination equal to the receiving user-id.

In cases where multiple messages or datasets are waiting to be received by the same user-id, all messages/datasets will be received

in one go by a single RECEIVE command. When a message is received, it will be deleted from the message queue (spool, punch, class).

This means that in cases where incoming mail should be preserved or data in incoming mail should be processed, some special action involving post-processing of the mail must be performed. Special action must be taken to support the intermix of mail messages and datasets.

Incoming mail can be saved from the RECEIVE command in two ways:

- 1 Allocate DDNAME SYSTSPRT to a dataset; messages and mail text will be saved to this dataset and can later be post-processed.

This dataset will have the following format:

```
READY
RECEIVE USERID(userid)
INMR901I Dataset ** MESSAGE ** from "sender" on "NODENAME"
text1 .....
text2 .....
INMR900I -----
---
INMR000I No more files remain for the receive command to process.
READY
END
```

- 2 Use RECEIVE subparameter LOGDATASET(dsname); the dataset 'dsname' will have the following format:

```
-----
RECEIVE          ** MESSAGE **                05 MAY 1997 15:13:11
  FROM:          sender's mail address        05 MAY 1997 15:12:11
text1 .....
text2 .....
-----
RECEIVE          ** MESSAGE **                05 MAY 1997 15:13:29
  FROM:          sender's mail address        05 MAY 1997 15:12:29
text1 .....
text2 .....
```

It should be noted that other types of mail could also exist, ie old-fashioned TSO mail sent directly via TRANSMIT and its acknowledgments. The routines must be able to handle these messages as well.

When one of the above methods has been used for receiving mail, the next step will be to extract the mailed data from the intermediate dataset.

The method actually selected for this mailbox system is the second method, using the logdataset parameter.

AUTOMATICALLY RECEIVING INCOMING MAIL

The principle of an automatic receive function is to receive mail in the mail receiver's mailbox as soon as mail arrives from SMTP to the JES spool and to notify the receiver of mail arrival.

We have investigated how this can be done, but not implemented the very best solution. For the time being, a batch job runs with a certain frequency executing SDSF in batch looking for output in class B, the output class selected for SMTP mail with jobid of SMTP; such output will have a destination equal to the mail receiver. This job will then receive the mail on behalf of the actual receiver and put their mail into their respective mailboxes, and send a message to the receivers. This implementation is described under SDSFBACK.

As far as we can see, SMTP itself does not have any such automatic receive/notify functions available. Another obvious possibility is some kind of exit. TSO has the Netmail exit INMXZ02, which will link into the TRANSMIT command and be triggered whenever mail is sent. Unfortunately, this exit will only have local impact, ie it has an effect only when mail is sent inside the same MVS system, and cannot be triggered when mail arrives from the outside world via SMTP. The JES2 exit 13 – Nje-mail exit – is a possibility, but again this exit will only be triggered when mail arrives as NJE-mail from another NJE-nodes, it will not be triggered when mail arrives as pure SMTP mail. JES2 exit 48 – SSI Sysout dataset unallocation exit – has not been tried, but it is believed that this exit could be a good candidate for implementing this function.

MAIL DELEGATION

In case a user is absent or for some other reason wants to delegate received mail, the mailbox system supports a delegation function in

which a user's received mail is sent to the appointed proxy(s).

The implementation is not fancy, no panel interface is implemented yet, but is simply based upon the presence of the member MAILROUT in the ISPF profile dataset belonging to the user who wants to have mail delegated. The member MAILROUT should contain a number of lines corresponding to the proxy users who are to receive the delegated mail, ie at least one record.

The format of the delegate record is:

```
proxy-user domain-of-proxy-user
```

or:

```
proxy-user@domain-of-proxy-user
```

The mail delegation function is handled by the CLIST MAILRECV.

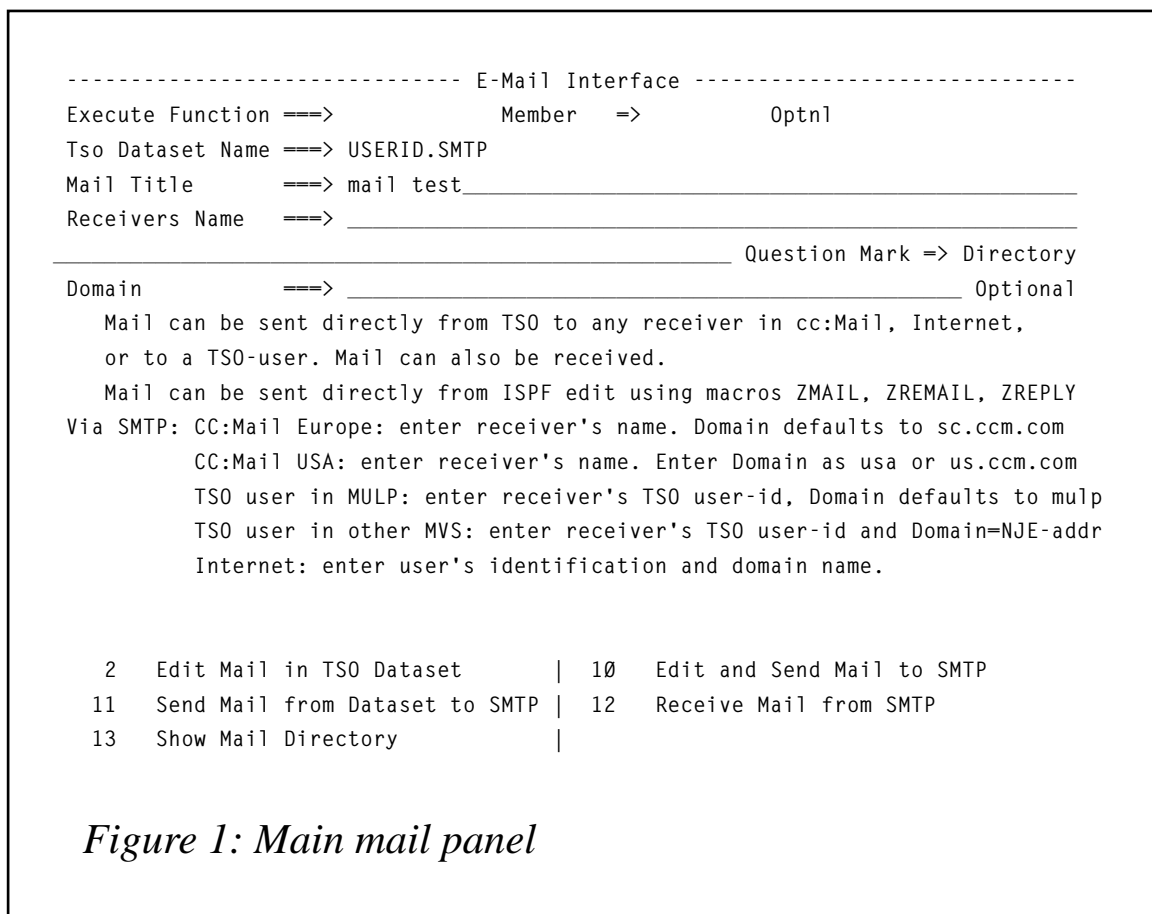
The main building bricks in the mailbox system consist of the following:

<i>Type</i>	<i>Name</i>	<i>Function</i>
CLIST	MAILDIRA	Build mail directory.
CLIST	MAILDIRS	Display mail directory.
CLIST	MAILEDIT	Edit mail in mailbox.
CLIST	MAILRECV	Receive mail and other data (datasets etc).
CLIST	MAILSENS	Build and send mail.
CLIST	TIMEOFFS	Get time-offset to GMT from SYS1.PARMLIB(CLOCKnn).
CLIST	USERNAME	Get name of current userid from RACF-definition.
CLIST	ZCC	Edit macro to build CC-receiver.
CLIST	ZMAIL	Edit macro to construct mail from dataset .
CLIST	ZNOMAIL	Edit macro to suppress mailing of mail.
CLIST	ZREMAIL	Edit macro to remail a mail.
CLIST	ZREPLY	Edit macro to construct reply to mail.
CLIST	ZSMTPC	Indirect edit macro to syntax check mail.
CLIST	ZSMTPI	Indirect edit macro to set ISPF notes in received.
CLIST	ZSMTPN	Indirect edit macro to set reply receivers.
CLIST	ZSMTPR	Indirect edit macro to save mail into mailbox.

Panel	INST81	Mailbox main ISPF panel.
Panel	INST81A	ISPF panel for support of ZMAIL edit macro.
Panel	INST81B	ISPF panel for support of ZCC edit macro.
Panel	INST81C	ISPF panel for displaying mail directory.
ispfmsgs	INST340-349	ISPF messages used in mailbox system.
skeleton	MAILDIR	ISPF skeleton for printing mail directory.
ispf-table	MAILDIR	Mail directory (ISPF table).
ispf-cmd	GETMAIL	ISPF command to receive mail.

Plus the following more general functions already available in this installation:

Editmacro	ZDELETE	Edit macro to delete current member.
Utility	ADSPNM	Get name of address space in CLIST/ISPF variable.
Utility	ASVTFIND	Determine presence of named address space.
Utility	ASVTSCAN	Scan ASVT for address spaces.



Utility	BREAK	Turn on terminal break mode.
Utility	HALT	Sleep command that can be terminated by modify.
Utility	INS025M	Copy dataset utility.
Utility	INS070	General subroutine to create CLIST/ISPF variable.
Utility	INTSYSIN	Create card image dataset from EXEC PARM-field.
Utility	NOBREAK	Turn off terminal break mode.
Utility	SLEEP	Set address space in wait for the same no of seconds.
Utility	SMFID	Get SMF-id in CLIST/ISPF variable.
Utility	TSOLINE1	A gentle CLEAR screen TSO command.
Macro	JOBNAME	Macro to get current address space name.

```

File Edit Confirm Menu Utilities Compilers Test Help
-----
EDIT      USERID.USERID.T112957.MAILFILE.OUTLIST      Columns 00001 00072
Command ==>          Scroll ==> PAGE
***** ***** Top of Data *****
==MSG>  ----- Send mode -----
==MSG> To send mail when editing finished: hit END (PF3/PF15).
==MSG> To send mail but stay in edit mode: enter ZREMAIL and hit Enter.
==MSG> To suppress and save or not-save mail: use ZNOMAIL or ZNOMAIL CAN.
==MSG> To enter CC-address use command ZCC evt followed by Receiver and domain.
- - - - - 6 Line(s) not Displayed
000007 To:      userid@mulp.pmvs01.com
000008 Subject: mail test
==MSG> Insert your data here
000009
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
***** ***** Bottom of Data *****

Figure 2: Building mail

```

CLIST	COMPLIB	CLIST to compress PDS.
CLIST	EDITRECV	General CLIST to test for pending edit recovery.
CLIST	GETMAIL	CLIST to create ISPF command GETMAIL.
CLIST	ISPFPCPY	CLIST to copy dataset using ISPF functions.
CLIST	SDSFBACK	CLIST to run SDSF in batch and parse SDSF panels.
CLIST	WAITINTJ	CLIST to wait until an address space arrives.
Procedure	ISPFBAT	JCL procedure to run ISPF in batch.

Figure 1 shows the main mail panel when entering the mail system.

Figure 2 shows how mail is built (selection 10) and shown to the user, including ISPF notes lines indicating possible actions.

Figure 3 shows how mail is built (selection 10) after a reset command showing all SMTP mail lines.

```

File Edit Confirm Menu Utilities Compilers Test Help
-----
EDIT      USERID.USERID.T112957.MAILFILE.OUTLIST      Columns 00001 00072
Command ==>          Scroll ==> PAGE
***** ***** Top of Data *****
000001 helo mulp.pmvs01.com
000002 mail from:<userid@mulp.pmvs01.com>
000003 rcpt to:<userid@mulp.pmvs01.com>
000004 data
000005 Date:      12 Dec 1997 11:30:01 +0100
000006 From:      userid@mulp.pmvs01.com (SURNAME GIVENNAME)
000007 To:        userid@mulp.pmvs01.com
000008 Subject:   mail test
000009
000009
000010 This is the mail text.
000011 This is the mail text.
000012
***** ***** Bottom of Data *****

Figure 3: Building mail after a reset command

```

```

File Edit Confirm Menu Utilities Compilers Test Help
-----
EDIT          USERID.SMTP(MAILTE10) - 01.00          Columns 00001 00072
Command ==>          Scroll ==> PAGE
***** ***** Top of Data *****
==MSG> To reply:          enter ZREPLY and hit ENTER.
==MSG> To save mail:      hit END (PF3/PF15).
==MSG> To forward mail:  enter ZMAIL and hit ENTER.
==MSG> To delete mail:   enter ZDELETE and hit ENTER.
000001  ** MESSAGE **          12 DEC 1997 11:33:41
000002  PMVS01  SMTP          12 DEC 1997 11:31:54
000003  Received: from mulp.pmvs01.com by mulp.pmvs01.com (IBM MVS SMTP V3R1)
000004    with BSMTp id 8327; Fri, 12 Dec 97 11:31:52 CET
000005  Date:          12 Dec 1997 11:31:43 +0100
000006  From:          userid@mulp.pmvs01.com (SURNAME GIVENNAME)
000007  To:            userid@mulp.pmvs01.com
000008  Subject:       mail test
000009
000010  This is the mail text.
000011  This is the mail text.
000012
***** ***** Bottom of Data *****

```

Figure 4: Example received mail

Figure 4 shows what received mail will look like, with ISPF note lines indicating possible actions.

Figure 5 shows how the mail directory is displayed (reduced contents).

```

----- Mail Directory ----- Row 543 to 566 of 797
Command ==>          SCROLL ==> PAGE
Command P to Print, U USERID Search for User-id, N NAME Search for Name,
D DOMAIN Search for Domain, F STRING Find string, X STRING Find all strings.
SORT: Sort on USERID, DOMAIN or NAME ascending A or descending D.
S Userid Domain Name          C
_ xxxx  xxxxx.dk          xxx yyyyyy
_ xxxx_xxxx          xxxxxx.com          xxxx zzzz
_ xxxlne mulp.pmvs01.com          xxxx wwww
_ xxxlnw mulp.pmvs01.com          xxxx vvvv
_ xxxluz mulp.pmvs01.com          xxxx zzzz
_ xxxluz pmvs01t xxxx kkkk
_ xxxxxx mulp.pmvs01.com          xxxx yyyy

```

Figure 5: Mail directory

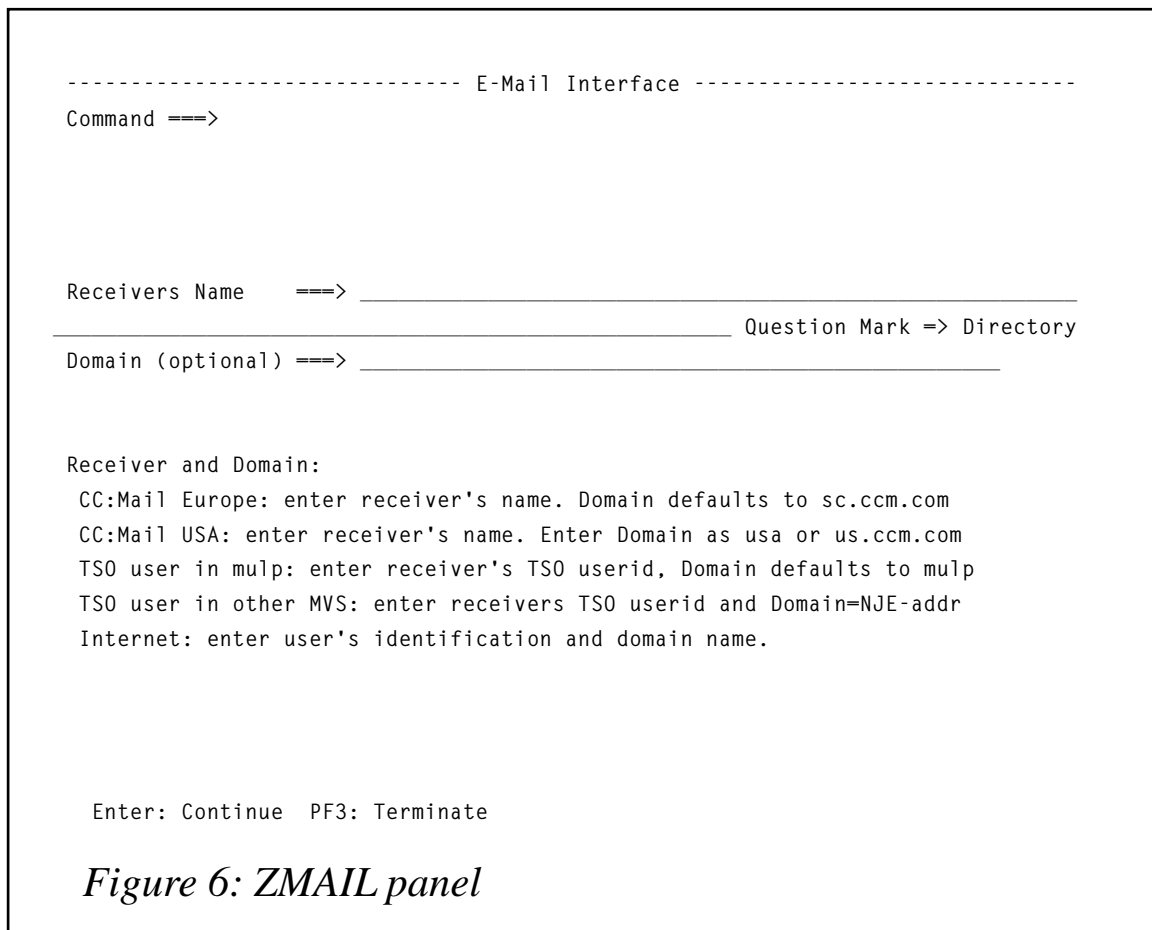


Figure 6 shows the panel displayed when command ZMAIL without receiver is called.

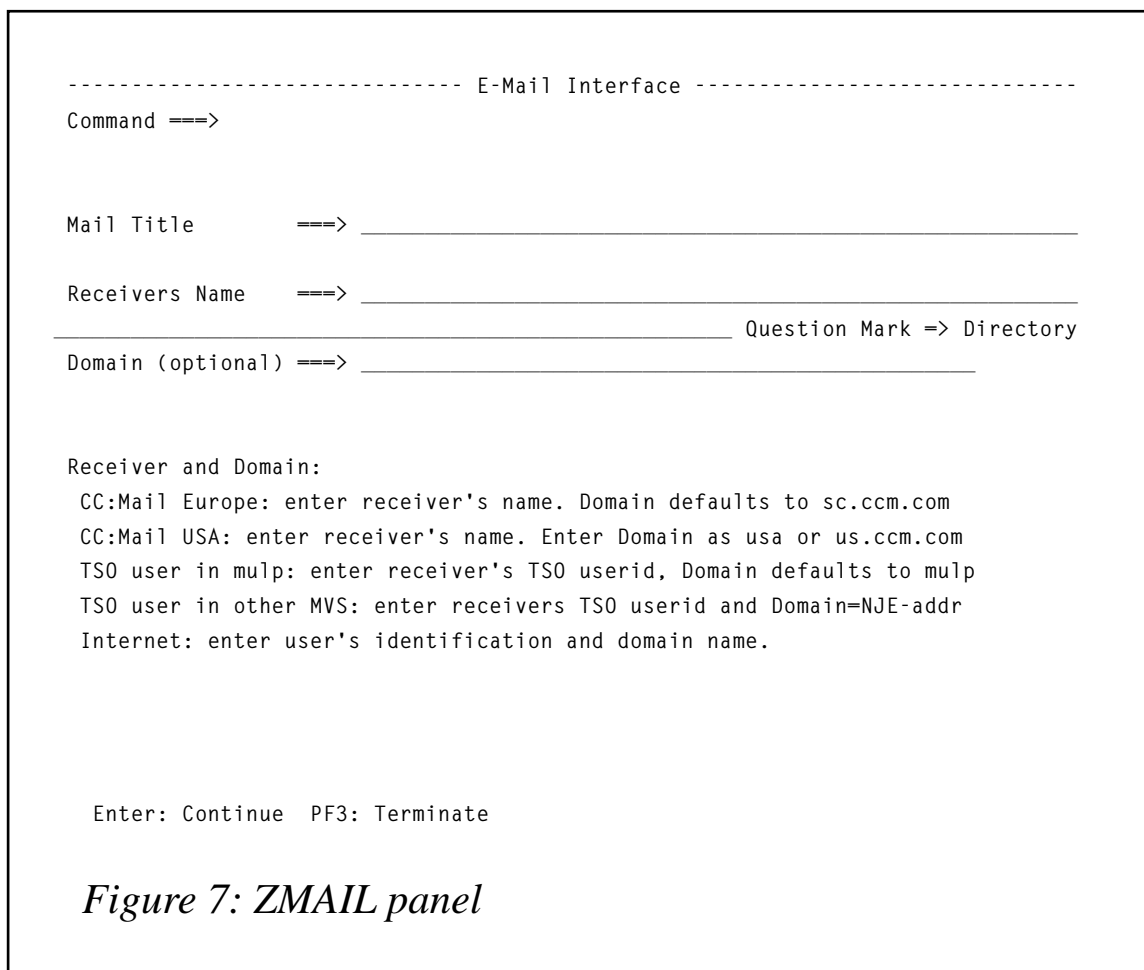
Figure 7 shows a similar panel displayed when command ZMAIL without receiver is called.

Figure 8 shows what the mailbox looks like under ISPF edit.

THE MAIL DIRECTORY

This mailbox system implements a mail directory based on an ISPF table. The table can be displayed, searched, sorted, printed, and main receiver and CC-receivers can be copied from the displayed table directly to the mail in the correct SMTP syntax.

The table is initially built on a list of local TSO users, since all TSO users in the system by default have a mailbox named 'userid at domainname', where domainname is the domain name for the executing



MVS system as described in the TCP/IP definitions in TCPIP.DATA parameters HOSTNAME and DOMAINORIGIN. Other users in other mail systems can be added *ad hoc* using the MAILDIRA CLIST described later. The table can be created initially using ISPF commands directly, for instance under option 7.6 test dialog services, eg:

```
TBCREATE MAILDIR KEYS(NAME USERID DOMAIN) WRITE SHARE
ISPEXEC TBCLOSE MAILDIR
```

For the purpose of adding users to the mail directory, a CLIST MAILDIRA is provided. The CLIST can be customized to take input in different formats; this implementation will primarily use the format as created using the print command via the skeleton MAILDIR. It is also designed to use a format created by the RACF utility product Vanguard.

The structure of the MAILDIR table, as displayed in ISPF function 7.4.5, is shown in Figure 9.

```

Menu  Functions  Utilities  Help
-----
EDIT      USERID.SMTP  Row 00209 of 00589
Command ==>          Scroll ==> PAGE
  Name          VV MM  Created   Changed   Size  Init  Mod  ID
ISPES00        01.00 97/12/18 97/12/18 20:02  106  106   0 RECEIVE
IPADRE50       01.00 97/12/17 97/12/17 19:02  358  358   0 RECEIVE
JANUAR10       01.00 97/12/17 97/12/17 19:02   49   49   0 RECEIVE
IPADRE10       01.00 97/12/16 97/12/16 09:15   17   17   0 MAILED
IPADRE20       01.00 97/12/16 97/12/16 11:01   37   37   0 RECEIVE
IPADRE30       01.00 97/12/16 97/12/16 16:01  346  346   0 RECEIVE
IPADRE40       01.00 97/12/16 97/12/16 16:11  359  359   0 REPLY
INOPRE00       01.00 97/12/15 97/12/15 09:02   38   38   0 RECEIVE
IPADRE00       01.00 97/12/15 97/12/15 13:57  196  196   0 MAILED
JANUAR00       01.00 97/12/12 97/12/12 12:48   15   15   0 SAVED
INSAML00       01.00 97/12/09 97/12/09 14:15   15   15   0 SAVED
ISPF4210       01.00 97/11/13 97/11/13 09:30   15   15   0 MAILED
INMXZ070       01.00 97/11/06 97/11/06 18:54   54   54   0 REPLY
INMXZ040       01.00 97/11/03 97/11/03 12:04   44   44   0 RECEIVE
INMXZ050       01.00 97/11/03 97/11/03 14:09   58   58   0 REPLY
INMXZ060       01.00 97/11/03 97/11/03 19:01   26   26   0 RECEIVE
ISPFV400       01.00 97/10/23 97/10/23 13:41   25   25   0 MAILED
ISPFV410       01.00 97/10/23 97/10/23 15:54   42   42   0 RECEIVE
ISPF4200       01.00 97/10/22 97/10/22 09:32   13   13   0 MAILED
JCLSF000       01.00 97/10/07 97/10/07 12:45  278  278   0 RECEIVE
ISPF00         01.00 97/09/24 97/09/24 14:01   21   21   0 RECEIVE
JOBIP00        01.00 97/09/16 97/09/16 15:33   31   31   0 RECEIVE
JOBPUG00       01.00 97/07/25 97/07/25 10:01   36   36   0 RECEIVE
JOBPUG10       01.00 97/07/25 97/07/25 11:53   50   50   0 RECEIVE
INSTDL00       01.00 97/06/26 97/06/26 10:40   29   29   0 RECEIVE
JCLCHA00       01.00 97/06/25 97/06/25 18:06   37   37   0 RECEIVE

```

Figure 8: Mailbox edit

For the purpose of displaying the mail directory, the CLIST MAILDIRS can be invoked. The invocation will normally be done as a subfunction from the mailbox system which can interface to the directory. It is also possible to produce a display-only by calling this CLIST directly from TSO.

MAILDIR

ISPF skeleton MAILDIR is used to print the mail directory.

```

+----- Dialog Test -----
|  Menu  Utilities  Help
|-----
|      Structure of Table MAILDIR      Table MAILDIR opened
| Command ==>          Scroll ==> PAGE
|
| Number of keys . . : 3                Number of rows . . . : 798
| Number of names . : 1                Current row pointer . : 0
| KEYS:      USERID      DOMAIN      NAME
|
| NAMES:      C
| ***** Bottom of data *****
|
+-----

```

Figure 9 MAILDIR table

```

***** MAIL DIRECTORY *****
)BLANK 1
)TB 27 54
USERID !DOMAIN!NAME
)DOT &TABLEID
&USERID!&DOMAIN!&NAME
)ENDDOT
)BLANK 1
***** E N D   O F   L I S T *****

```

OUTPUTLIST

An example of an outputlist from the print command via skeleton MAILDIR is shown below. The format can be used as input to MAILDIRA.

```

xxxx                xxxxx.dk                xxx yyyyyy
xxxx_xxxx           xxxxxx.com                xxxx zzzz
xxxln               mulp.pmvs01.com             xxxx www
xxxlnw              mulp.pmvs01.com             xxxx vvvv
xxxluz              mulp.pmvs01.com             xxxx zzzz
xxxluz              pmvs01t xxxx kkkk
xxxxxx              mulp.pmvs01.com             xxxx yyyy

```

MAILDIRA

The CLIST MAILDIRA is used to build (add users to) the mail directory.


```

/*          */
/* MAILDIRA
/* Update SMTP Mail Directory from flatfile.
/* Called directly under TSO
/*
/* Parameters:
/* HOSTNAME: Current Host Name; MUST BE mulp.pmvs01.com
/*           as defined in INST.TCPIP.TCPIP.DATA.SMTP in
/*           HOSTNAME+DOMAINORIGIN
/* USERS   : input dataset name containing user definitions
/* SRC     : MAILDIR list of users as created by MAILDIRS print
/*           function via skeleton:
/*           pos 2-26 userid
/*           pos 28-53 domain
/*           pos 55-80 name (surname given-name)
/*           VANGUARD list of TSO-users created by VANGUARD from
/*           RACF TSO segment:
/*           pos 26-32 userid (TSO-userid)
/*           pos 2-24 name (surname given-name)
/*           Domain defaults to HOSTNAME
/*
/* ISPF TABLE USED: MAILDIR
/*
/* Utilities used:
/* SLEEP
/*          */
PROC 0 DEBUG(NEBUG) +
USERS(your.dsname(MAILDIR)) SRC(MAILDIR) +
HOSTNAME(mulp.pmvs01.com)
CONTROL NOMSG NOFLUSH NOLIST NOCONLIST NOSYMLIST NOCAPS
ATTN DO
  SET &FLUSH = FLUSH          /* NEXT STATEMENT MUST BE NULL LINE    */

END
ERROR DO
  SET &RET = &LASTCC
  RETURN
END
SET &RET = 0
IF &SYSCAPS(&STR(&DEBUG)) = DEBUG THEN DO
  CONTROL MSG NOFLUSH LIST CONLIST SYMLIST
  WRITE =====> Entering &SYSICMD <=====
END
IF &FLUSH = FLUSH THEN DO
  ISPEXEC TBEND &TABLEID
  ISPEXEC LIBDEF ISPTLIB
  ISPEXEC LIBDEF ISPTABL
  CLOSFILE USERS
  FREE FI(USERS)
  SET &DEBUG = NEBUG
  ISPEXEC VPUT (DEBUG) SHARED

```

```

SET &ZEDSMMSG = &str(Function interrupted)
ISPEXEC SETMSG MSG(ISRZ001)
EXIT CODE(4)
END
IF &SYSISPF = &STR(NOT ACTIVE) THEN DO
  WRITE =====> Sorry only executable under ISPF (&SYSICMD).
  EXIT CODE(16)
END
ISPEXEC CONTROL ERRORS RETURN
ISPEXEC VGET (ZSCREEN)
SET COPIED = NO
SET &TABLEID = &STR(MAILDIR)
SET &TLIB = INST.ISPTLIB
ISPEXEC VGET (ZSCREEN)
ISPEXEC LIBDEF ISPTLIB DATASET ID('&TLIB')
ISPEXEC LIBDEF ISPTABL DATASET ID('&TLIB')
ISPEXEC TBCLOSE &TABLEID
ALLOC FI(USERS) DA('&USERS') SHR REUSE
CLOSFILERS USERS
SET &RET = 0
OPENFILE USERS INPUT
SET &OPENRET = &RET
IF (&SYSDSN(&STR('&TLIB(&TABLEID)')) NE OK OR &OPENRET > 0 THEN DO
  ISPEXEC CONTROL DISPLAY LINE START(14)
  SLEEP 1
  WRITE =====> +
  &STR(Table &TLIB(&TABLEID) &SYSDSN(&STR('&TLIB(&TABLEID)')))
  WRITE =====> +
  &STR(Users &USERS &SYSDSN(&STR('&USERS')))
  ISPEXEC LIBDEF ISPTLIB
  ISPEXEC LIBDEF ISPTABL
  CLOSFILERS USERS
  FREE FI(USERS)
  EXIT CODE(16)
END
SET &ORDER = &STR()
ISPEXEC TBOPEN &TABLEID WRITE SHARE
SET &SORTCRIT = &STR(NAME,C,A,USERID,C,A,DOMAIN,C,A)
ISPEXEC TBSORT &TABLEID FIELDS(&SORTCRIT)
SET &MAXREAD = 4096
SET &RET = 0
DO WHILE &RET NE 400 AND &READ < &MAXREAD
  SET &READ = &READ + 1
  SET &RET = 0
  GETFILE USERS
  IF &RET = 0 THEN DO
    IF &SRC = VANGUARD THEN DO
      SET &STRING = &SYSLC(&SUBSTR(26:32,&STR(&SYSNSUB(1,&USERS))))
      SET &RSTRING = &STR() /* CLEAR FOR CALL TO SUBROUTINE RETURN */
      SET &RET = 0
      SYSCALL VARSTRNG RSTRING STRING(&STR('&SYSNSUB(1,&STRING)')) +

```

```

DEBUG(&DEBUG)
/* SYSCALL RETURN CODE DOES NOT TAKE ERROR ROUTINE */
SET &RET = &LASTCC
SET &USERID = &STR(&SYSNSUB(1,&RSTRING))
/*      */
SET &STRING = &SYSLC(&STR(&SYSNSUB(1,&HOSTNAME)))
SET &RSTRING = &STR() /* CLEAR FOR CALL TO SUBROUTINE RETURN */
SET &RET = 0
SYSCALL VARSTRNG RSTRING STRING(&STR('&SYSNSUB(1,&STRING)')) +
DEBUG(&DEBUG)
/* SYSCALL RETURN CODE DOES NOT TAKE ERROR ROUTINE */
SET &RET = &LASTCC
SET &DOMAIN = &STR(&SYSNSUB(1,&RSTRING))
/*      */
SET &STRING = &SYSLC(&SUBSTR(2:24,&STR(&SYSNSUB(1,&USERS))))
SET &RSTRING = &STR() /* CLEAR FOR CALL TO SUBROUTINE RETURN */
SET &RET = 0
SYSCALL VARSTRNG RSTRING STRING(&STR('&SYSNSUB(1,&STRING)')) +
DEBUG(&DEBUG)
/* SYSCALL RETURN CODE DOES NOT TAKE ERROR ROUTINE */
SET &RET = &LASTCC
SET &NAME = &STR(&SYSNSUB(1,&RSTRING))
/*      */
SET &C = &STR()
END
IF &SRC = MAILDIR THEN DO
SET &STRING = &SYSLC(&SUBSTR(2:26,&STR(&SYSNSUB(1,&USERS))))
SET &RSTRING = &STR() /* CLEAR FOR CALL TO SUBROUTINE RETURN */
SET &RET = 0
SYSCALL VARSTRNG RSTRING STRING(&STR('&SYSNSUB(1,&STRING)')) +
DEBUG(&DEBUG)
/* SYSCALL RETURN CODE DOES NOT TAKE ERROR ROUTINE */
SET &RET = &LASTCC
SET &USERID = &STR(&SYSNSUB(1,&RSTRING))
/*      */
SET &STRING = &SYSLC(&SUBSTR(28:53,&STR(&SYSNSUB(1,&USERS))))
SET &RSTRING = &STR() /* CLEAR FOR CALL TO SUBROUTINE RETURN */
SET &RET = 0
SYSCALL VARSTRNG RSTRING STRING(&STR('&SYSNSUB(1,&STRING)')) +
DEBUG(&DEBUG)
/* SYSCALL RETURN CODE DOES NOT TAKE ERROR ROUTINE */
SET &RET = &LASTCC
SET &DOMAIN = &STR(&SYSNSUB(1,&RSTRING))
/*      */
SET &STRING = &SYSLC(&SUBSTR(55:80,&STR(&SYSNSUB(1,&USERS))))
SET &RSTRING = &STR() /* CLEAR FOR CALL TO SUBROUTINE RETURN */
SET &RET = 0
SYSCALL VARSTRNG RSTRING STRING(&STR('&SYSNSUB(1,&STRING)')) +
DEBUG(&DEBUG)
/* SYSCALL RETURN CODE DOES NOT TAKE ERROR ROUTINE */
SET &RET = &LASTCC

```

```

    SET &NAME = &STR(&SYSNSUB(1,&RSTRING))
/*          */
    SET &C      = &STR()
END
IF &SYSCAPS(&STR(&SYSNSUB(1,&USERID))) NE &STR(USERID ) THEN DO
IF &SYSCAPS(&SUBSTR(1:10,&STR(&SYSNSUB(1,&NAME)))) NE +
&STR(#####) THEN DO /* do not include invalid names */
    IF &STR(&SYSNSUB(1,&USERID)) NE &STR() AND +
&STR(&SYSNSUB(1,&DOMAIN)) NE &STR() THEN DO
        SET &MAXCNT = 64
        SET &CNT = 0
        SET &RET = 0
        ISPEXEC TBADD &TABLEID &ORDER MULT(512)
        DO WHILE &RET = 8 AND &CNT < &MAXCNT
            SET &CNT = &CNT + 1
            SET &RET = 0
            ISPEXEC TBADD &TABLEID &ORDER MULT(512)
        END
        IF &RET = 8 THEN DO
            WRITE &STR(&SYSNSUB(1,&USERID)) not Added.
        END
        ELSE DO
            WRITE &STR(&SYSNSUB(1,&USERID)) Added.
        END
    END
END
END
END
END
END
ISPEXEC TBSORT &TABLEID FIELDS(&SORTCRIT)
ISPEXEC TBCLOSE &TABLEID
ISPEXEC TBEND &TABLEID
ISPEXEC LIBDEF ISPTLIB
ISPEXEC LIBDEF ISPTABL
CLOSFILE USERS
FREE FI(USERS)
EXIT CODE(0)
/*          */
/*  INLINE SUBROUTINES          */
/*          */
VARSTRNG: +
PROC 1 RSTRING STRING() DEBUG(NEBUG)
/*          */
/*  INLINE VARSTRNG ROUTINE; TRUNCATE TRAILING BLANKS          */
/*          */
CONTROL NOMSG NOFLUSH NOLIST NOCONLIST NOSYMLIST
ATTN DO
    SET &FLUSH = FLUSH /* NEXT STATEMENT MUST BE NULL LINE */

END
ERROR DO

```

```

SET &RET = &LASTCC
RETURN
END
IF &SYSCAPS(&STR(&DEBUG)) NE DEBUG THEN DO
  ISPEXEC VGET (DEBUG)
END
IF &STR(&DEBUG) = &STR() THEN DO
  SET &DEBUG = NEBUG
END
IF &SYSCAPS(&STR(&DEBUG)) = DEBUG THEN DO
  CONTROL MSG NOFLUSH LIST CONLIST SYMLIST
END
IF &FLUSH = FLUSH THEN DO
  SET &DEBUG = NEBUG
  ISPEXEC VPUT (DEBUG) SHARED
  SET &ZEDSMMSG = &str(Function interrupted)
  ISPEXEC SETMSG MSG(ISRZ001)
  RETURN CODE(0)
END
ISPEXEC CONTROL ERRORS RETURN
SYSREF &RSTRING
SET &RSTRING = &STR(&SYSNSUB(1,&STRING))
IF &STR(&SYSNSUB(1,&STRING)) = &STR() THEN DO
  RETURN CODE(0)
END
SET &RET = 0
SET &ALLBLANKS = +
&STR(
    -
    -
    -
    )
IF &LENGTH(&SYSNSUB(1,&STRING)) > 0 THEN DO
  SET &SRCHSPC = &STR( ) /* look for last blank */
  SET &STARTSPC = 1
  SET &LOCSPC = 1
  SET &LASTSPC = 0
  SET &LENRCP = &LENGTH(&STR(&SYSNSUB(1,&STRING)))
  SET &MAXT = &LENRCP
  SET &R = 0
  DO WHILE &LOCSPC > 0 AND &STARTSPC <= &LENRCP AND &R < &MAXT
    SET &R = &R + 1
    SET &LOCSPC = +
    &SYSINDEX(&STR(&SRCHSPC),&STR(&SYSNSUB(1,&STRING)),&STARTSPC)
    IF &LOCSPC > 0 THEN DO
      SET &STARTSPC = &LOCSPC + 1
      SET &LASTSPC = &LOCSPC
      IF &LASTSPC > 0 AND &LENRCP > &EVAL(&LASTSPC + 1) +
      THEN DO /* REST BLANKS */
        IF &SUBSTR(&LASTSPC:&LENRCP,&STR(&SYSNSUB(1,&STRING))) = +
        &SUBSTR(1:&LENRCP-&LASTSPC-1,&STR(&ALLBLANKS)) THEN DO
          SET &STRING = +

```

```

        &SUBSTR(1:&LASTSPC-1,&STR(&SYSNSUB(1,&STRING)))
        SET &R = &MAXT
    END
END
END
END
END
SET &RSTRING = &STR(&SYSNSUB(1,&STRING))
RETURN CODE(0)
END
/*          */

```

Editor's note: this article will be continued in the next issue.

Nils Plum
Systems Programmer (Denmark)

© Xephon 1998

Link between NCPs

We have ten computers running under MVS/ESA 5.2.0 connected via SNA with X.25. Each computer has its own subarea. We defined switched links (CVC) between each computer. An INN link is defined in the NCP (G12INNX on the MCH12 and G00INNA on the MCH0). A VTAM switched node is defined for each computer on the network. To manage these INN links easily, I wrote two REXX CLISTs that run under NetView. They are called ACTINN and INACTINN, and are used to establish or to stop a connection.

The commands are executed under NetView: ACTINN xx to establish a connection; and INACTINN xx to stop a connection – where xx is the suffix of the switched node for VTAM.

ACTINN

```

/*REXX*/
IF ARG(1)≠'' THEN DO
    IF ARG(1)=ALL THEN DEPTS="31 33 35 37 38 44 51 59 94 76"
    ELSE DEPTS=ARG(1)
    DO I=1 TO WORDS(DEPTS)
        DEPT=WORD(DEPTS,I)
        NOEUD='X'DEPT'CVC'
        SAY '** ACTIVATION MSNF CVC NOEUD "NOEUD" **'
    END
END

```

```

"TRAP AND SUPPRESS MESSAGES IST093I,IST453I,IST116I,IST061I"
"V NET,ACT,ID="NOEUD
"WAIT FOR MESSAGES"
"MSGREAD"
IF MSGID()=="IST093I" THEN DO
  DO WHILE MSGVAR(1)≠NOEUD
    "WAIT CONTINUE"
    "MSGREAD"
  END
  IF DEPT="DIAL" THEN PU_NOEUD="P"DEPT"IN0"
  ELSE PU_NOEUD="P"DEPT"CVC"
  SAY "* DIAL PU:" PU_NOEUD
  "TRAP AND DISPLAY MESSAGES IST607I,IST241I,IST464I,IST493I"
  "VARY NET,DIAL,ID="PU_NOEUD
  "WAIT FOR MESSAGES"
  "MSGREAD"
END
ELSE SAY "ATTENTION ERROR, DEPARTMENT" DEPT "NOT AVAILABLE"
END
SAY "END OF ACTIVATION"
END

```

INACTINN

```

&CONTROL ERR
&IF .&1 = . &THEN &GOTO -ERRP1
&IF .&1 = .ALL &THEN &GOTO -ALL
&NUM = &1
&ID0 = &CONCAT X &1
&ID0 = &CONCAT &ID0 CVC
&ID1 = &CONCAT P &1
&ID1 = &CONCAT &ID1 CVC0
&ID2 = &CONCAT P &1
&ID2 = &CONCAT &ID2 CVC1
&ID3 = &CONCAT P &1
&ID3 = &CONCAT &ID3 CVC2
&ID4 = &CONCAT P &1
&ID4 = &CONCAT &ID4 CVC3
&WRITE * DESACTIVATION MSNF CVC *
VARY NET,HANGUP,ID=&ID1
VARY NET,HANGUP,ID=&ID2
VARY NET,HANGUP,ID=&ID3
VARY NET,HANGUP,ID=&ID4
VARY NET,INACT,ID=&ID0,F
&EXIT
-ERRP1
&WRITE PARAMETRE MANQUANT
&EXIT
-ALL

```

```
VARY NET,INACT,ID=X31CVC,FORCE
VARY NET,INACT,ID=X33CVC,FORCE
VARY NET,INACT,ID=X35CVC,FORCE
VARY NET,INACT,ID=X37CVC,FORCE
VARY NET,INACT,ID=X38CVC,FORCE
VARY NET,INACT,ID=X44CVC,FORCE
VARY NET,INACT,ID=X51CVC,FORCE
VARY NET,INACT,ID=X59CVC,FORCE
VARY NET,INACT,ID=X76CVC,FORCE
VARY NET,INACT,ID=X94CVC,FORCE
&EXIT
```

NCP

```
PRINT NOGEN
*
* ATTENTION IDBLKC 062 MCH3 ENQUETE INN *
*****
* ATTENTION , IF YOU CHANGE THE NCP NAME *
* YOU MUST CHANGE THE OTHER PARAMETERS. *
*****
*
* OPTIONS USERGEN=(CBEX25,ATFTUNE), *
* NEWDEFN=(YES,,SUPP) *
*** ACF/VTAM ONLY - PCCU FOR ES/9000 VTAM *
*
PCCU650 PCCU CUADDR=650, *
MAXDATA=4300, *
AUTODMP=NO, DEFAULT=NO *
AUTOIPL=NO, DEFAULT= NO *
AUTOSYN=YES, LINK TO VFYLM AVOIDS AUTOMATIC LOAD *
BACKUP=YES, *
DUMPDS=DUMP3745, DDNAME OF DUMP FILE *
CDUMPDS=CDMP3745, DDNAME OF CSP FILE *
MDUMPDS=MDMP3745, DDNAME OF MOSS FILE *
NETID=CP, *
OWNER=VTAM69, *
INITEST=NO, *
SUBAREA=31, VTAM *
VFYLM=NO LINK TO AUTOSYN (CI-DESSUS) *
* STATOPT='PCCU 650 3745 ' *
*** BUILD ***
*****
NCP692 BUILD BFRS=240, (SEE HOST UNITSZ) DEFAUT=60 *
BRANCH=100, DEFAUT=100 *
CATRACE=(YES,10), *
CSMHDR=27F5C711, CRITIC MSG 3270 BSC *
CSMHDR=40E3C5E7E3405C5C, EXTRA TO WHAT'S GONE BEFORE *
CSMSG=C5D5C7D6D9C7C5D4C5D5E340C4E440D9C5E2C5C1E4, *
```



```

CSMSGC=6040C1D5E240828587A4954B, *
CWALL=26, *
DR3270=YES, *
LOADLIB=LOAD3745, VTAM DDNAME POUR LOADMODULE NCP *
MAXSSCP=2, *
MAXSUBA=63, *
MEMSIZE=4M, *
MODEL=3745, *
NETID=CP, *
NEWNAME=NCP692, *
NUMHSAS=5, IN REL 3 TAKE CARE IF 3745 ADDED *
PRTGEN=NOGEN, *
PUNAME=NCP692, *
QUALIFY=NCP740, QUALIFIER FOR NCP FILE *
SUBAREA=33, (MVS LYON 31 ,3745 33,TEST 32 ) *
TRACE=(YES,64), *
TRANSFR=18, NB BUFFERS FOR NCP *
TRCPIU=2000, DEFAULT PIU PR TRACE LINE & PATH *
TYPGEN=NCP, *
TYP SYS=OS, *
ATF.USGTIER=3, (3745 MOD 210) *
USGTIER=3, *
NTUNECOL=YES, *
VERSION=V7R4, FOR NCP *
VRPOOL=30, DEFAULT V5 = 768 *
X25.IDNUMH=0, *
X25.PREFIX=X, *
X25.USGTIER=2, *
X25.SNAP=YES, *
X25.MCHCNT=4, *
X25.MAXPIU=4K *
* STATOPT='NCPV5 -- 3745 '
* NPA=(YES,DR), NETWORK PERFORMANCE ANALYSER
* SESSACC=(YES,ALL,,50000000,1000), BYTES AND PIU COUNT
* X25.BYTTHROD=65536, NPM ACCOUNTIING, NOT INSTALLED
* X25.SEGTHROD=65536, NPM ACCOUNTIING, NOT INSTALLED
*****
*** SYSCNTRL ***
*****
DYNCTRL SYSCNTRL OPTIONS=(BHSASSC,MODE,RCNTRL,RCOND,RECMD,RIMM,NAKLIM,*
SESSION,SSPAUSE,XMTLMT,STORDSP)
*****
*** HOST ES/9000 ***
*****
HOSTEXP HOST MAXBFRU=15, MESSAGE VTAM MAX BEFORE 16 4300/292 *
UNITSZ=292, 292 X 16 = 4762 OCTETS *
BFRPAD=0, NECESSARY FOR OS *
SUBAREA=31, DEFAULT *
INBFRS=12 CODE WITH LINE CA IN VERSION 5
*****

```

```

* PUDRPOOL /LUDRPOOL/ *
*****
      PUDRPOOL NUMBER=5
      LUDRPOOL NUMTYP1=20,NUMTYP2=50
*****
*   CHEMINS DE LYON VERS L'EXTERIEUR *
*****
*
D69EXP  PATH DESTSA=31,ER0=(31,1),ER1=(31,1),ER2=(31,1),ER3=(31,1), *
          ER4=(31,1),VR0=0,VR1=1,VR2=2,VR3=3, *
          VRPWS00=(5,20),VRPWS01=(5,128),VRPWS02=(5,64), *
          VRPWS10=(5,20),VRPWS11=(5,128),VRPWS12=(5,64), *
          VRPWS20=(5,20),VRPWS21=(5,128),VRPWS22=(5,64), *
          VRPWS30=(5,20),VRPWS31=(5,128),VRPWS32=(5,64)
D69TEST PATH DESTSA=32,ER0=(32,1),ER3=(32,1),ER4=(32,1)
CRETEIL PATH DESTSA=(1,2,3,35,36,37),ER0=(36,1),ER1=(37,1), *
          ER2=(37,1),ER3=(36,1)
D76E    PATH DESTSA=(22,52,53),ER0=(52,1),ER1=(53,1), *
          ER2=(53,1),ER3=(52,1)
D33E    PATH DESTSA=(14,54,55),ER0=(54,1),ER1=(55,1), *
          ER2=(54,1),ER3=(55,1)
D38E    PATH DESTSA=(20,56,57),ER0=(56,1),ER1=(57,1), *
          ER2=(56,1),ER3=(57,1)
D37E    PATH DESTSA=(10,38,12),ER0=(38,1),ER1=(12,1), *
          ER2=(38,1),ER3=(12,1)
D44E    PATH DESTSA=(5,4,60),ER0=(4,1),ER1=(60,1), *
          ER2=(4,1),ER3=(60,1)
D51E    PATH DESTSA=(16,50,51),ER0=(50,1),ER1=(51,1), *
          ER2=(50,1),ER3=(51,1)
RENNES  PATH DESTSA=(25,26,27),ER0=(27,1),ER1=(27,1), *
          ER2=(27,1),ER3=(27,1)
LILLE   PATH DESTSA=(28,29,30),ER0=(30,1),ER1=(30,1),ER2=(30,1), *
          ER3=(30,1)
TOULOUSE PATH DESTSA=(7,8,9,58,59),ER0=(58,1),ER1=(59,1), *
          ER2=(58,1),ER3=(59,1)
*
*****
*   GROUPE DE LINES S D L C *
*****
*
G690SDLC GROUP CLOCKNG=EXT, X
          DUPLEX=FULL, X
          ISTATUS=ACTIVE, X
          LNCTL=SDLC, X
          NEWSYNC=NO, X
          NRZI=NO, YES IF CERTAIN MODEMS NON IBM X
          PAUSE=0.2, X
          REPLYTO=0.5, 0.270 + PAUSE X
          RETRIES=(3,3,2), 2 SEQ OF 3 EVERY 3 SEC. X
          SERVLIM=4, X
          SSCPFM=USSCS, MESS. SNA NON STANDARD X

```

```

USSTAB=USSTAB3, X
DLOGMOD=M32782, X
VPACING=3, X
PACING=1, X
TYPE=NCP, X
MODETAB=SLC32XX
STATOPT='GROUP LN SDLC '
*
*
*-----*
* LIGNE S N A C. I. C. S. R. E. E. L. *
* CONTROLEUR E : 15 SCREENS 1 PRINTER *
*-----*
L690000E LINE ADDRESS=(26,HALF), LINE SDLC (3174) *
SERVLIM=17, *
ISTATUS=ACTIVE, *
SPEED=9600 DOES NOTHING (CLOCKNG=EXT)
* STATOPT='SDLC*LIGNE 26 HALFDUPLEX'
C690000E PU ADDR=01, *
DISCNT=NO, PU REMAINS ACTIVE WITHOUT LU CONNECTED *
MAXDATA=521, MAX DATA PER PIU X
MAXOUT=4, MAX NO. OF PIU EMITTED WITHOUT REPLY X
PASSLIM=7, X
PUDR=YES, X
LOGAPPL=A695CICS, X
PUTYPE=2, X
VPACING=4 ENTER HOST AND 3745
* STATOPT='PU MANAGEMENT E '
T690000EA LU LOCADDR=02, *
LUDR=YES
* STATOPT='PU MANAGEMENT E '
T690000EB LU LOCADDR=03, *
LUDR=YES
* STATOPT='PU MANAGEMENT E '
T690000EC LU LOCADDR=04, *
LUDR=YES
* STATOPT='PU MANAGEMENT E '
T690000ED LU LOCADDR=05, *
LUDR=YES
* STATOPT='PU MANAGEMENT E '
T690000EE LU LOCADDR=06, *
LUDR=YES
* STATOPT='PU MANAGEMENT E '
T690000EF LU LOCADDR=07, *
LUDR=YES
* STATOPT='PU MANAGEMENT E '
T690000EG LU LOCADDR=08, X
LUDR=YES
* STATOPT='PU MANAGEMENT E '
T690000EH LU LOCADDR=09, *
LUDR=YES
* STATOPT='PU MANAGEMENT E '

```

```

T69000EI LU      LOCADDR=10,
                  LUDR=YES
*
* STATOPT='PU MANAGEMENT E '
T69000EJ LU      LOCADDR=11,
                  LUDR=YES
*
* STATOPT='PU MANAGEMENT E '
T69000EK LU      LOCADDR=12,
                  LUDR=YES
*
* STATOPT='PU S*MANAGEMENT E '
T69000EL LU      LOCADDR=13,
                  LUDR=YES
*
* STATOPT='PU S*MANAGEMENT E '
T69000EM LU      LOCADDR=14,
                  LUDR=YES
*
* STATOPT='PU S*MANAGEMENT E '
T69000EN LU      LOCADDR=15,
                  LUDR=YES
*
* STATOPT='PU S*MANAGEMENT E '
T69000EO LU      LOCADDR=16,
                  LUDR=YES
*
* STATOPT='PU S*MANAGEMENT E '
T69000EP LU      LOCADDR=17,
                  LUDR=YES
*
* STATOPT='PU *MANAGEMENT E '
*-----*
*  LIGNE S N A      C. I. C. S.  R. E. E. L.
*  CONTROLEUR B : 15 SCREENS  1 PRINTER
*-----*
L690000B LINE    ADDRESS=(9,HALF), LINE SDLC (3174)
                  SERVLIM=17,
                  ISTATUS=ACTIVE,
                  SPEED=9600      DOES NOTHING (CLOCKNG=EXT)
*
* STATOPT='SDLC*LIGNE 9 HALF DUPLEX'
C690000B PU      ADDR=01,
                  DISCNT=NO,      PU REMAINS ACTIVE WITHOUT LU CONNECTED
                  MAXDATA=521,    MAX DATA PER PIU
                  MAXOUT=4,      MAX NO. OF PIU EMITTED WITHOUT REPLY
                  PASSLIM=7,
                  PUDR=YES,
                  LOGAPPL=A695CICS,
                  PUTYPE=2,
                  VPACING=4      ENTER HOST AND 3745
*
* STATOPT='PU MANAGEMENT E '
T690000BA LU     LOCADDR=02,
                  LUDR=YES
*
* STATOPT='PU MANAGEMENT B '
T690000BB LU     LOCADDR=03,
                  LUDR=YES
*
* STATOPT='PU MANAGEMENT B '
T690000BC LU     LOCADDR=04,
                  LUDR=YES

```

```

*          STATOPT='PU MANAGEMENT B '
T69000BD LU LOCADDR=05,
LUDR=YES
*          STATOPT='PU MANAGEMENT B '
T69000BE LU LOCADDR=06,
LUDR=YES
*          STATOPT='PU MANAGEMENT B '
T69000BF LU LOCADDR=07,
LUDR=YES
*          STATOPT='PU MANAGEMENT B '
T69000BG LU LOCADDR=08,
LUDR=YES
*          STATOPT='PU MANAGEMENT B '
T69000BH LU LOCADDR=09,
LUDR=YES
*          STATOPT='PU MANAGEMENT B '
T69000BI LU LOCADDR=10,
LUDR=YES
*          STATOPT='PU MANAGEMENT B '
T69000BJ LU LOCADDR=11,
LUDR=YES
*          STATOPT='PU MANAGEMENT B '
T69000BK LU LOCADDR=12,
LUDR=YES
*          STATOPT='PU B*MANAGEMENT B '
T69000BL LU LOCADDR=13,
LUDR=YES
*          STATOPT='PU B*MANAGEMENT B '
T69000BM LU LOCADDR=14,
LUDR=YES
*          STATOPT='PU B*MANAGEMENT B '
T69000BN LU LOCADDR=15,
LUDR=YES
*          STATOPT='PU B*MANAGEMENT B '
T69000BO LU LOCADDR=16,
LUDR=YES
*          STATOPT='PU B*MANAGEMENT B '
T69000BP LU LOCADDR=17,
DLOGMOD=SCS3287,MODETAB=BIND3287,
PACING=7,
LUDR=YES
*          STATOPT='IMP *MANAGEMENT B '
*-----*
*          LIGNE S N A      C. I. C. S.  R. E. E. L.
*          CONTROLEUR N : 15 SCREENS 1 PRINTER COMPTA
*-----*
L690000N LINE ADDRESS=(4,HALF), LINE SDLC (3174)
SERVLIM=17,
ISTATUS=ACTIVE,
SPEED=9600      DOES NOTHING (CLOCKNG=EXT)
*          STATOPT='SDLC*LIGNE 4 HALF DUPLEX'

```

C690000N	PU	ADDR=01,		*
		DISCNT=NO,	PU REMAINS ACTIVE WITHOUT LU CONNECTED	*
		MAXDATA=521,	MAX DATA PER PIU	X
		MAXOUT=4,	MAX NO. OF PIU EMITTED WITHOUT REPLY	X
		PASSLIM=7,		X
		PUDR=YES,		X
		LOGAPPL=A695CICS,		X
		PUTYPE=2,		X
		VPACING=4	ENTER HOST AND 3745	
*		STATOPT='PU MANAGEMENT E	'	
T690000NA	LU	LOCADDR=02,		*
		LUDR=YES		
*		STATOPT='PU MANAGEMENT N	'	
T690000NB	LU	LOCADDR=03,		*
		LUDR=YES		
*		STATOPT='PU MANAGEMENT N	'	
T690000NC	LU	LOCADDR=04,		*
		LUDR=YES		
*		STATOPT='PU MANAGEMENT N	'	
T690000ND	LU	LOCADDR=05,		*
		LUDR=YES		
*		STATOPT='PU MANAGEMENT N	'	
T690000NE	LU	LOCADDR=06,		*
		LUDR=YES		
*		STATOPT='PU MANAGEMENT N	'	
T690000NF	LU	LOCADDR=07,		*
		LUDR=YES		
*		STATOPT='PU MANAGEMENT N	'	
T690000NG	LU	LOCADDR=08,		X
		LUDR=YES		
*		STATOPT='PU MANAGEMENT N	'	
T690000NH	LU	LOCADDR=09,		*
		LUDR=YES		
*		STATOPT='PU MANAGEMENT N	'	
T690000NI	LU	LOCADDR=10,		*
		LUDR=YES		
*		STATOPT='PU MANAGEMENT N	'	
T690000NJ	LU	LOCADDR=11,		*
		LUDR=YES		
*		STATOPT='PU MANAGEMENT N	'	
T690000NK	LU	LOCADDR=12,		*
		LUDR=YES		
*		STATOPT='PU S*MANAGEMENT N	'	
T690000NL	LU	LOCADDR=13,		*
		LUDR=YES		
*		STATOPT='PU S*MANAGEMENT N	'	
T690000NM	LU	LOCADDR=14,		*
		LUDR=YES		
*		STATOPT='PU S*MANAGEMENT N	'	
T690000NN	LU	LOCADDR=15,		*
		LUDR=YES		

```

*          STATOPT='PU S*MANAGEMENT N'
T69000NO LU LOCADDR=16,
LUDR=YES
*          STATOPT='PU S*MANAGEMENT N'
T69000NP LU LOCADDR=17,
LUDR=YES
*          STATOPT='PU *MANAGEMENT N'
*
*-----*
*   LIGNE S N A      C. I. C. S.  R. E. E. L.
*   CONTROLEUR C : 15 SCREENS  1 PRINTER  SALLE C
*-----*
L69000C LINE ADDRESS=(6,HALF), LINE SDLC (3174)
SERVLIM=17,
ISTATUS=ACTIVE,
SPEED=9600      DOES NOTHING (CLOCKNG=EXT)
*          STATOPT='SDLC*LIGNE 6 HALF DUPLEX'
C69000C PU  ADDR=01,
DISCNT=NO,      PU REMAINS ACTIVE WITHOUT LU CONNECTED
MAXDATA=521,    MAX DATA PER PIU
MAXOUT=4,       MAX NO. OF PIU EMITTED WITHOUT REPLY
PASSLIM=7,
PUDR=YES,
LOGAPPL=A695CICS,
PUTYPE=2,
VPACING=4      ENTER HOST AND 3745
*          STATOPT='PU MANAGEMENT C '
T69000CA LU LOCADDR=02,
LUDR=YES
*          STATOPT='PU MANAGEMENT C '
T69000CB LU LOCADDR=03,
LUDR=YES
*          STATOPT='PU MANAGEMENT C '
T69000CC LU LOCADDR=04,
LUDR=YES
*          STATOPT='PU MANAGEMENT C '
T69000CD LU LOCADDR=05,
LUDR=YES
*          STATOPT='PU MANAGEMENT C '
T69000CE LU LOCADDR=06,
LUDR=YES
*          STATOPT='PU MANAGEMENT C '
T69000CF LU LOCADDR=07,
LUDR=YES
*          STATOPT='PU MANAGEMENT C '
T69000CG LU LOCADDR=08,
LUDR=YES
*          STATOPT='PU MANAGEMENT C '
T69000CH LU LOCADDR=09,
LUDR=YES
*          STATOPT='PU MANAGEMENT C '

```

```

T69000CI LU      LOCADDR=10,
                  LUDR=YES
*
* STATOPT='PU MANAGEMENT C '
T69000CJ LU      LOCADDR=11,
                  LUDR=YES
*
* STATOPT='PU MANAGEMENT C '
T69000CK LU      LOCADDR=12,
                  LUDR=YES
*
* STATOPT='PU C*MANAGEMENT C '
T69000CL LU      LOCADDR=13,
                  LUDR=YES
*
* STATOPT='PU C*MANAGEMENT C '
T69000CM LU      LOCADDR=14,
                  LUDR=YES
*
* STATOPT='PU C*MANAGEMENT C '
T69000CN LU      LOCADDR=15,
                  LUDR=YES
*
* STATOPT='PU C*MANAGEMENT C '
T69000CO LU      LOCADDR=16,
                  LUDR=YES
*
* STATOPT='PU C*MANAGEMENT C '
T69000CP LU      LOCADDR=17,
                  DLOGMOD=SCS3287,MODETAB=BIND3287,
                  PACING=7,
                  LUDR=YES
*
* STATOPT='IMP *MANAGEMENT C '
*
*-----*
*   LIGNE S N A      C. I. C. S.  R. E. E. L.
*   CONTROLEUR D : 15 SCREENS  1 PRINTER  SALLE D
*-----*
L690000D LINE     ADDRESS=(7,HALF), LINE SDLC (3174)
                  SERVLIM=17,
                  ISTATUS=ACTIVE,
                  SPEED=19200      DOES NOTHING (CLOCKNG=EXT)
*
* STATOPT='SDLC*LIGNE 7 FULL DUPLEX 19200'
C690000D PU       ADDR=01,
                  DISCNT=NO,      PU REMAINS ACTIVE WITHOUT LU CONNECTED
                  MAXDATA=521,    MAX DATA PER PIU
                  MAXOUT=4,      MAX NO. OF PIU EMITTED WITHOUT REPLY
                  PASSLIM=7,
                  PUDR=YES,
                  LOGAPPL=A695CICS,
                  PUTYPE=2,
                  VPACING=4      ENTER HOST AND 3745
*
* STATOPT='PU MANAGEMENT D '
T69000DA LU      LOCADDR=02,
                  LUDR=YES
*
* STATOPT='PU MANAGEMENT D '
T69000DB LU      LOCADDR=03,
                  LUDR=YES

```



```

*          STATOPT='PU MANAGEMENT D '
T690000DC LU LOCADDR=04,
LUDR=YES
*          STATOPT='PU MANAGEMENT D '
T690000DD LU LOCADDR=05,
LUDR=YES
*          STATOPT='PU MANAGEMENT D '
T690000DE LU LOCADDR=06,
LUDR=YES
*          STATOPT='PU MANAGEMENT D '
T690000DF LU LOCADDR=07,
LUDR=YES
*          STATOPT='PU MANAGEMENT D '
T690000DG LU LOCADDR=08,
LUDR=YES
*          STATOPT='PU MANAGEMENT D '
T690000DH LU LOCADDR=09,
LUDR=YES
*          STATOPT='PU MANAGEMENT D '
T690000DI LU LOCADDR=10,
LUDR=YES
*          STATOPT='PU MANAGEMENT D '
T690000DJ LU LOCADDR=11,
LUDR=YES
*          STATOPT='PU MANAGEMENT D '
T690000DK LU LOCADDR=12,
LUDR=YES
*          STATOPT='PU D*MANAGEMENT D '
T690000DL LU LOCADDR=13,
LUDR=YES
*          STATOPT='PU D*MANAGEMENT D '
T690000DM LU LOCADDR=14,
LUDR=YES
*          STATOPT='PU D*MANAGEMENT D '
T690000DN LU LOCADDR=15,
LUDR=YES
*          STATOPT='PU D*MANAGEMENT D '
T690000DO LU LOCADDR=16,
LUDR=YES
*          STATOPT='PU D*MANAGEMENT D '
T690000DP LU LOCADDR=17,
DLOGMOD=SCS3287,MODETAB=BIND3287,
PACING=7,
LUDR=YES
*          STATOPT='IMP *MANAGEMENT D '
*
*-----*
*   LIGNE S N A      C. I. C. S.  R. E. E. L.
*   CONTROLEUR A : 15 SCREENS  1 PRINTER  SALLE A
*-----*
L6900000A LINE ADDRESS=(8,HALF), LINE SDLC (3174)

```

```

SERVLIM=17, *
ISTATUS=ACTIVE, *
SPEED=9600 DOES NOTHING (CLOCKNG=EXT)
* STATOPT='SDLC*LIGNE 8 HALF DUPLEX'
C690000A PU ADDR=01, *
DISCNT=NO, PU REMAINS ACTIVE WITHOUT LU CONNECTED *
MAXDATA=521, MAX DATA PER PIU X
MAXOUT=4, MAX NO. OF PIU EMITTED WITHOUT REPLY X
PASSLIM=7, X
PUDR=YES, X
LOGAPPL=A695CICS, X
PUTYPE=2, X
VPACING=4 ENTER HOST AND 3745
* STATOPT='PU MANAGEMENT D '
T690000AA LU LOCADDR=02, *
LUDR=YES
* STATOPT='PU MANAGEMENT D '
T690000AB LU LOCADDR=03, *
LUDR=YES
* STATOPT='PU MANAGEMENT D '
T690000AC LU LOCADDR=04, *
LUDR=YES
* STATOPT='PU MANAGEMENT D '
T690000AD LU LOCADDR=05, *
LUDR=YES
* STATOPT='PU MANAGEMENT D '
T690000AE LU LOCADDR=06, *
LUDR=YES
* STATOPT='PU MANAGEMENT D '
T690000AF LU LOCADDR=07, *
LUDR=YES
* STATOPT='PU MANAGEMENT D '
T690000AG LU LOCADDR=08, X
LUDR=YES
* STATOPT='PU MANAGEMENT D '
T690000AH LU LOCADDR=09, *
LUDR=YES
* STATOPT='PU MANAGEMENT D '
T690000AI LU LOCADDR=10, *
LUDR=YES
* STATOPT='PU MANAGEMENT D '
T690000AJ LU LOCADDR=11, *
LUDR=YES
* STATOPT='PU MANAGEMENT D '
T690000AK LU LOCADDR=12, *
LUDR=YES
* STATOPT='PU S*MANAGEMENT D '
T690000AL LU LOCADDR=13, *
LUDR=YES
* STATOPT='PU S*MANAGEMENT D '
T690000AM LU LOCADDR=14, *

```

```

*          LUDR=YES
*          STATOPT='PU S*MANAGEMENT D'
T690000AN LU          LOCADDR=15,          *
*          LUDR=YES
*          STATOPT='PU S*MANAGEMENT D'
T690000A0 LU          LOCADDR=16,          *
*          LUDR=YES
*          STATOPT='PU S*MANAGEMENT D'
T690000AP LU          LOCADDR=17,          *
*          DLOGMOD=SCS3287,MODETAB=BIND3287, X
*          PACING=7,                      X
*          LUDR=YES
*          STATOPT='IMP *MANAGEMENT D'
*
*-----*
*   LIGNE S N A      C. I. C. S.   R. E. E. L.      *
*   CONTROLEUR L : 15 SCREENS  1 PRINTER  SALLE A  *
*-----*

```

Editor's note: this article will be continued in the next issue.

Claude Dunand (France)

© Xephon 1998

Networking experts

Why not share your expertise and earn money at the same time? *TCP/SNA Update* is looking to publish technical articles about your experience with TCP/IP, SNA, and connecting the enterprise environment. We are looking for information about problems and solutions you have experienced that would be of value to other networking specialists. We will publish your article (after vetting by our expert panel) and send you a cheque when the article is published. Why not call now for a free copy of our *Notes for contributors*?

TCP/SNA news

Wall Data has announced Version 5.2 of its Rumba Office 95/NT connectivity suite that provides access to IBM mainframe, AS/400, VAX, Unix, and HP hosts, as well as corporate intranets, extranets, and the Internet.

Version 5.2 now includes the Rumba Internet Companion, a connectivity application said to simplify on-line access and help users organize public and corporate information. Other new features include AS/400 display, print, file transfer, shared folders, and data access functionality over IBM's native TCP/IP.

There's also support of NFS Client 95/NT to speed access and transfer of Unix-based files and applications over TCP/IP.

For further information contact:
Wall Data, 11332 Northeast 122nd Way,
Kirkland, WA 98034-6931, USA.
Tel: (206) 814 9255.
Wall Data (UK) Ltd, Wall Data House, 418
Bath Road, Longford, West Drayton,
Middx, UB7 0EA, UK.
Tel: (0181) 476 5000.

* * *

NetManage has begun shipping a licensed version of SNA Server 3270 Applet which provides basic 3270 emulation for SNA Server connected users, including browser-based access to IBM hosts. NetManage will release an updated applet once Version 4.0 of SNA Server ships, and will include support for running the applet from within Internet Explorer. Available for Windows 3.x, 95, and NT desktops, the applet supports SNA Server Versions 2.11, 3.0, and 4.0.

For further information contact:
NetManage, 10725 N De Anza Blvd,
Cupertino, CA 95014, USA.
Tel: (408) 973 7171.

NetManage UK, Chameleon House, 22
Frederick Sanger Road, Guildford GU2
5YD, UK.
Tel: (01483) 302333.

* * *

Banyan has unveiled Version 8.0 of its TCP/IP-based Vines network operating system. Among the new version features are LDAP support, better messaging capabilities, and a new version of the StreetTalk Explorer GUI management tool.

Improved TCP/IP server-to-server software is for enabling multiple Vines and StreetTalk for Windows NT servers to communicate with each other using TCP/IP.

For further information contact:
Banyan Systems, 120 Flanders Rd,
Westborough, MA 01581-5013, USA.
Tel: (508) 898 1000.
Banyan Systems (UK), Banyan House,
Northwood Park, Gatwick Road, Crawley,
W Sussex, RH10 2XN, UK.
Tel: (01293) 612284.

* * *

FTP Software has announced Version 2.0 of its Firefox Internet Gateway Suite for NetWare, claimed to be the first NetWare-based SPX/IPX to TCP/IP gateway to offer support for NT Version 4.0 NetWare clients. It also supports NetWare clients running Windows 95, 3.1, DOS, and Macintosh.

The suite is designed to provide NetWare users with managed and secure access to TCP/IP networks while reducing administration and maintenance.

For further information contact:
FTP Software, 100 Brickstone Square, 5th
Floor, Andover, MA 01810, USA.
Tel: (508) 685 4000.



xephon