

The Case for *iSeries* and Thin Client Computing

By *Thibault Dambrine*

Introduction

If someone asked you right now what are the basic working principles of “thin client architecture”, would you have an answer? “Citrix”, “MetaFrame” and “Citrix Server” are names and terms you may be familiar with but could you say much more about them? Perhaps you even have heard about the term RDP but did you know thin client support is one of Microsoft’s fastest growing markets?

Until a few months ago, I too, had no idea myself about Citrix or RDP. I could not have answered these questions. Recently, however, I did go through a software implementation that involved both an iSeries and a Citrix Server. While working on this assignment, I also learned about Microsoft’s Remote Desktop Protocol, or RDP. Sometimes, describing the practical application of a new technique better describes how it works than any technical spec sheet. In this article,

I will share with you my experience, and what I have learned. I will first describe the type of company I work for and I will describe the project that gave me a taste of what thin client architecture is all about. The implementation I will describe involved using an iSeries computer and the Citrix thin client product. To give some perspective, I will also compare Citrix’s MetaFrame with Microsoft’s RDP.

As a final introductory note, I hope this material will give the reader an indication of the potential this technology offers to help resolve situations where client/server computing in remote locations is a challenge.

Background

I currently work as a contractor for a large integrated company located in Calgary, Alberta. For the purpose of this article, I will name it “CanOilCo”. The company is well known across the country for its petroleum products and its distribution network, which spans all major airports as well as a strong retail presence with its gas stations across the country.

The Alberta Energy and Utilities Board, AEUB for short, has for its mission to “To ensure that the discovery, development, and delivery of Alberta’s resources take place in a manner that is fair, responsible, and in the public interest”. Consistent with their mission, the AEUB requires all oil and gas producers, such as CanOilCo to produce very detailed reporting data on all the components that come out of a well, at the well head. Gas for example, also comes with sulfur and natural gas liquids, which include, amongst others, ethane, butane, propane and condensate. Each of these components are reported and accounted for. Dealing with oil and gas, one deals also with mineral rights and royalties. These too have to be accounted for. In this industry, this set of specialized measuring tasks is referred to as “Production Accounting”.

To handle its Production Accounting, the team I work with at CanOilCo uses a software package called PraccSoft, built by PraccSoft Inc. - a local software company. This package was originally developed on AS/400 technology, using a traditional RPG/5250 emulation “green screen” or “character display” type of user interface. While for many, “green screen” is synonymous with old technology, it does have its advantages. CanOilCo’s iSeries system is located in Calgary. Its oil and gas producing facilities



Thibault Dambrine

are scattered in many remote areas of the province. Often, it is very difficult to get any more than a 56KB line to a remote facility, especially in the far north and in the mountainous Foothills area to the West. In such situations, green screen user interfaces are ideal: with at most 24x80 characters or less to transmit per screen, it is very economical on bandwidth. Even a 56KB line can handle comfortably up to 3 remote 5250 screens with very little visible communication delay.

One of the common complaints iSeries-based software vendors hear from customers is that their software is “proprietary”. Translation: It can run only on iSeries and not on any other machines. To break free from this predicament, many traditional iSeries-based software development companies are creating new, platform-independent versions of their formerly iSeries-only software.

In the case of the PraccSoft production accounting package, PraccSoft Inc. has created a new, client-server version of their package using a development tool called PROGRESS®. The PROGRESS-written clients run on Windows PCs while the server database can be based on anything from iSeries to UNIX/Oracle to SQL Server.

The platform-independent nature of the PROGRESS-written client allows PraccSoft Inc. to sell their software to a much broader range of customers, many of whom would not have previously considered their software. In effect, the same software package can be configured for to Two-Man-Band Exploration running on a Microsoft SQL server as well as CanOilCo running an iSeries model 840. With this shift from iSeries-only to Client-Server configuration, PraccSoft Inc. announced last year that they would no longer support the “pure” iSeries solution anymore. The team I work with at CanOilCo has been an PraccSoft Inc. Software client for several years already, using the AS/400-only version of PraccSoft. At this point, CanOilCo had a choice to upgrade the AS/400-only version of Praccsoft to the client-server version or to pick a different production accounting software package altogether. After careful consideration, CanOilCo opted to upgrade.

The Project

The “project”, as it is described in this article, consisted in upgrading the PraccSoft production accounting system from “Green Screen” to “Client Server” using Citrix thin client. My responsibility in this project was to support and facilitate the technical part of this conversion. In this situation, the architecture of the new “GUI” product was pre-determined. Having no previous experience with the Citrix architecture, this was a learning experience for me.

If you could think of the new product in its simplest form, one PC client accessing the iSeries locally through a fast local area network, you could conceivably load the client software on this single client PC, load the database on the iSeries and be up and running fairly quickly. This system would probably work equally well with a small number of PC clients. The only inconvenience to the IT staff would be the maintenance, which

would require one to make upgrades to each client individually every time there is a change in the client software. The speed however, would not be an issue, as a 10 to 100 megabit Ethernet network would carry the client/server conversation.

The logical question that follows would be “what if you have 50 or 100 clients, in 20 different locations?” Doing manual upgrades for this type of setup could be terribly time consuming. Why go client/server at all if that is the type of headache this would bring? Viewed from this angle, Client-Server looks like a maintenance nightmare and despite the hype, a bit of a step backwards. As a base building block for this article, we have to define the term WTS. Windows Terminal Server, or WTS is an NT server operating system that allows Windows (TM)

multi-user remote capability and connectivity on Windows NT from different operating systems: Linux, Macintosh, Unix (Sun Solaris/x86, IBM AIX, SCO, Compaq Tru64, HP/UX), and Windows (95, 98, ME, NT 4.0 & 2000). Here is a high-level presentation of how ICA works. The Citrix Independent Computing Architecture, ICA model includes the following components:

- server software
- network protocol
- client software

On the server, ICA has the unique ability to separate an application’s logic from its user interface and transport the visible screen interface to the client device over a number of standard network protocols.

The ICA network protocol transports keystrokes, mouse-clicks and screen updates to the client, consuming less than 5 kilobits-per-second of network bandwidth. This efficiency enables the latest 32-bit applications to be accessed with exceptional performance from existing PCs, Windows-based terminals and other operating system clients. Citrix client side users can work with the application’s interface, sending keystrokes and mouse clicks over the network, and receiving screen updates from the server.

Some perspective on the Citrix Technology

The original thin Windows solution was Citrix’s WinFrame, based on a version of Windows NT 3.51 modified by Citrix to support multiple, simultaneous users. Microsoft licensed part of the technology behind WinFrame to create its Windows Terminal Server (WTS). Although Citrix still ships and supports WinFrame, it has created another product, called MetaFrame, that runs on top of WTS and enhances it. At the core of Citrix’s products – both WinFrame and MetaFrame – is the Independent Computing Architecture (ICA) protocol. The ICA protocol sits above the network transport layer. It can thus work over most networking transport technologies, including TCP/IP, IPX, NetBIOS, and PPP.

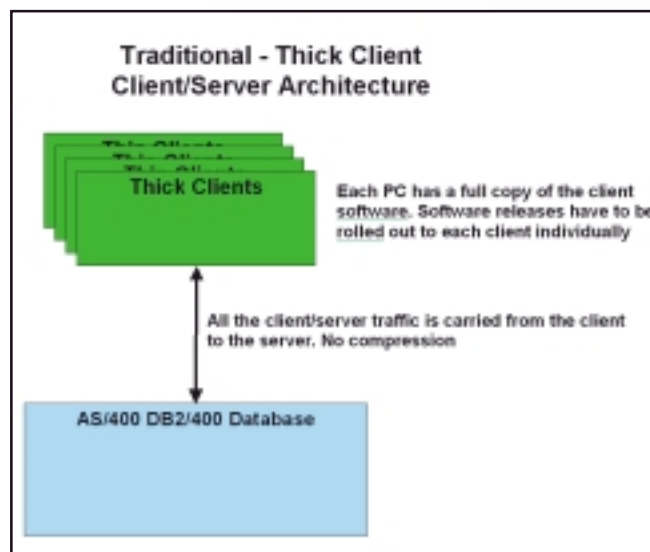


Figure 1

applications to be run on a server and displayed on many different operating systems, running on remote machines. Citrix technology uses WTS as a base. Enter the Thin Client architecture. In this implementation, CanOilCo used Citrix’s Independent Computing Architecture or ICA for short.

Microsoft Windows Terminal Server (WTS) – also known as Windows NT Terminal Server Edition (TSE) – coupled with Citrix MetaFrame Server offers server-based computing that enables

WinFrame uses ICA to communicate a highly optimised version of the screen activity of an application to the client, using compression and client-side caching of common Windows objects – cursors, icons and bitmaps – so applications appear responsive even over long-distance WANs or over a modem. (For more information on ICA, see Appendix 1.)

Figure 2 is a diagram showing how Thin Client architecture works using Citrix. Again, note that Citrix is not the only thin client product available on the market. Each remote client (also known as “thin client”) initiates a session with the Citrix Server. When the session is initiated, each thin client starts the designated application (in our case, the PraccSoft client) in a protected mode. In effect, a portion of the Citrix Server is allocated to their own particular client processing requirements. In this situation, there are two links or conversations going per thin client:

- (1) The conversation between the thin client (what the remote user sees on his/her screen) and the Citrix Server.
- (2) The conversation between the allocated application on the Citrix Server, who is itself a client and the database file server (in our situation, an iSeries).

The first conversation, labeled as (1), is designed to send only compressed screen images and retrieve key-strokes/mouse clicks from the thin client. It uses absolutely minimal bandwidth. This is ideal for situations where the thin client can only access the line with a limited bandwidth, such as 56 or 128KB. Note that in this situation, the signal is distributed through CanOilCo’s Intranet, but it can be also distributed via the public Internet also.

The second conversation, labeled as (2), has no such compression algorithm. It can be likened to any type of PC client, carrying on a normal conversation with the database server. In this case, SQL requests, ODBC calls, and heavier traf-

fic are common. Since this traffic travels over a much thicker pipe, as in a 100-megabit Ethernet line for example, the Citrix Server to Database Server conversation is fluid and unrestrained. In short, the Citrix Server performs all the computation and processing locally for each individual thin client and itself plays the actual “client” role in the “client/server” architecture. What the user sees and interacts with is the Thin client presentation layer of Citrix.

Benefits and Costs

To give a point of reference on the advantages to be gained by using this type of technology, here are some numbers, from the tests we have performed. We installed a complete copy of our client software on one of the PCs located at a

conversation takes place locally, over a 100-Megabit Ethernet line. The only traffic carried to the user on the 56KB line was compressed screen images going one way and keystrokes and mouse-clicks on the return trip. One advantage of this configuration, beyond the response time, is the ease of version upgrade for the Clients. In a Citrix Server/Thin Client situation, upgrades to the client software can be applied centrally, on the Citrix server, rather than on each desktop PC. In a thick client situation, for a client software upgrade, the rollout has to take place at each user workstation.

In the case of CanOilCo, my team took advantage of one more feature of this software. ICA thin client sessions can be conveniently launched from an intranet-based web page. Doing so, we eliminated the necessity to distribute and maintain conventional ICA desktop icons to start the thin client sessions for each PraccSoft user. All the users have to do is to visit the PraccSoft startup web page on the CanOilCo intranet, click on an icon and this is enough to start the thin client ICA session. There are benefits, and there are costs. Setting up a Citrix server is not free. You have to count the cost of a fully configured NT server, the cost of the Citrix software itself, and the extra initial effort. The price of these components may vary with the configurations but in all cases, one may expect to spend several thousand dollars in hardware and effort.

Obviously, the cost per user goes down, as more thin clients access the software.

Is Citrix technology economical? Is it what you would need? What are the considerations? If you have few remote users, the cost per user will be high. There will be a point where there will be too few users to justify the cost of a Citrix server. This point may vary from company to company. With a low number of users, the only justification for a Citrix server may simply be the more efficient use it makes of the scarce bandwidth resources, and that may be enough.

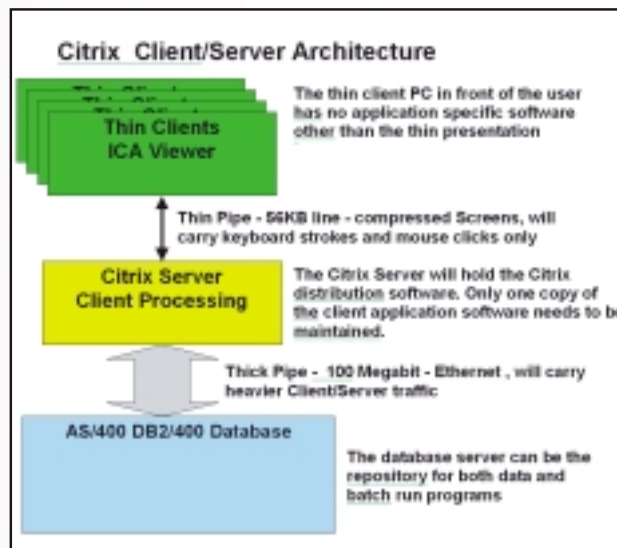


Figure 2

remote well site. This type of installation can also be described as “thick client”. A 56KB line served the remote location. In this configuration, each of the client requests and responses had to squeeze through the remote location’s 56KB line before getting to the server and the replies had to go through the same constraint. In this configuration, we measured responses between 50 and 70 seconds. By contrast, the exact same software installed on a Citrix server at the head-office, reached by the same remote well location using Citrix thin client yielded a 3-second response time for these same screens. The secret of success here is that all the heavy client/server



The distance between your clients and your server is also a consideration to examine. If your users are far-flung, having to send someone on a trip to maintain a thick client may be expensive. The heavier traffic generated on a long distance by a thick client to its server may also be an expense to consider. How widespread is ICA? Only Citrix Systems Inc. could answer this question with any precision. From my own experience, here are at least two very visible cases. They are also two completely different applications.

(1) The JD Edwards software company, is migrating its World ERP software to a client-server model. The name of their new software is called One-World®. Citrix ICA technology is commonly used to distribute screen images to thin client equipped JD Edwards One-World users.

(2) The Toronto User Group Association's information system: Currently, TUG is has purchased an association management software package named MTrack. This NT-based software, produced by a Vancouver company (MTrack Software Ltd.) can reside on a server located anywhere in the world. It can then be accessible by all its users using Citrix Thin clients over an Internet connection. In the proposed scenario, TUG's data would actually reside on a server somewhere on the west coast for cost advantage and proximity to the vendor. Your association manager and the TUG Board directors would access this server as thin clients via the Internet using Citrix Independent Computing Architecture (ICA).

A word about technical limitations:

On the high end, the official maximum clients we have been told could be handled by a Citrix server is 60 thin clients. Experience at CanOilCo says 20 is a more realistic value. This means that if you have a large number of clients who would need Citrix functionality, you may need many more NT-based Citrix servers than you perhaps expected initially. When one hears "NT" and "Mission Critical" in the same sentence, invariably, doubts spring to mind. NT is a good operating system, no question, but perhaps it may not be described as a paramount of reliability, if compared to an iSeries operating system. There is more on this topic in the next paragraphs.

While writing this piece, I also had the serendipitous opportunity to meet Dr Frank Soltis, the father of the S/38, the AS/400 and now the iSeries, in person. I will share with you some of his insights, and how Citrix could conceivably meet iSeries in more than one way.

Here is one experience he described. In a recent benchmarking exercise, two computers one Intel-based PC running NT and one an iSeries were pitted against each other for benchmark measurement. Both servers were running the same benchmarking application, both using the same version of NT. During this exercise, a surprising event was discovered. The exact same NT operating system, running the same set of programs and the exact same data, ran more reliably on the iSeries machine than on the PC. "How can this be explained?" he asked the audience.

In Dr Soltis's own words, when NT is about to crash, there are predictable states of "sickness" in the NT operating system. In some detectable and quantifiable way, NT says, "I am sick", then shows symptoms of "I am going to die" before finally crashing. When NT runs on an iSeries partition, the main or managing partition (OS/400) monitors for signs sickness or imminent death in the NT operating system data. When they appear, OS/400 rectifies the situation before these error situations further deteriorate and become deadly (read NT crashes). Having that extra safeguard on an iSeries partition explains the extra reliability NT gets on this platform. IBM helping Microsoft – who would have thought it?

Putting the pieces together, here is a recap the last few paragraphs:

- Citrix and RDP run on NT.
- The iSeries can support one or more copies of NT in one or more partitions.

Knowing these facts, one could dream up a client-server situation requiring an iSeries server and a number of NT server machines dishing out Citrix or RDP signals to thin clients. With the partitioning ability of the iSeries, the same scenario could be played with ONLY ONE iSeries computer, partitioned to run both the iSeries portion of the software and the thin client server on a separate partition of the same hardware.

Radius
INFORMATION SYSTEMS LTD.

2680 Skymark Ave
Suite # 420
Mississauga, ON
L4W 5L6
[T] 905.602.9772
[F] 905.602.9859
www.radiusinfo.com

a COGNICASE company

E-Business Solutions

ERP Integration
BPCS Specialists

Financial Brokerage Solutions

EDI Solutions

Business Intelligence Integration

AS/400 Technology Specialists

Our Services

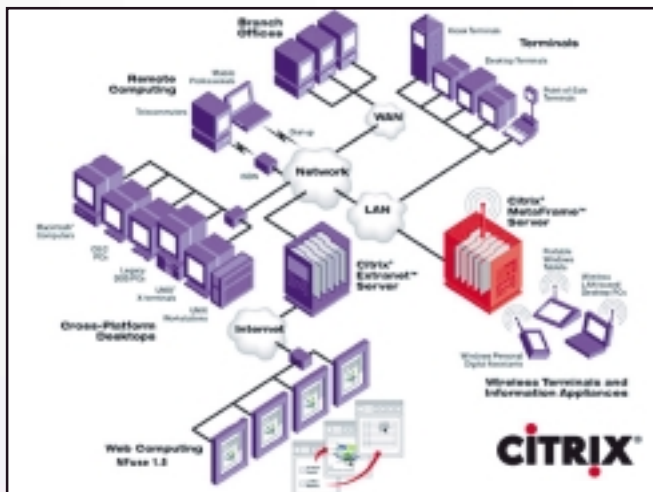
Market Choices and Thin Client Considerations: Citrix MetaFrame vs. Microsoft RDP

While Citrix is probably the best-known name in the “thin client computing software” supplier arena, there is a silent killer in these waters. On January 31 2001, in a situation reminiscent of the Netscape/Explorer market share battle, Microsoft announced that its Terminal Services-based thin client solution is the leading thin client OS, with 59 percent of the market. According to International Data Corporation (IDC) findings and Microsoft’s sales data, Terminal Services has more than doubled its share of the thin client market in the past year, and IDC expects this growth to continue.

Since Microsoft did not license ICA, it needed to create its own protocol so WTS would be useful without having to purchase Citrix’s product. Remote Desktop Protocol (RDP) is the result. It is closely related to the protocol used by Microsoft’s NetMeeting. Remote Desktop Protocol or RDP was first released under Windows NT® Server 4.0, Terminal Server Edition (TS 4.0). Like Citrix, the protocol was designed to provide remote display and input capabilities over network connections for Windows-based applications running on a server. At release time, the performance of the protocol, (RDP 4.0), was considered adequate for LAN connections, but slow for low-bandwidth connections such as the one described in this article. Since then, Microsoft has come up with RDP 5.0 and then with TSAC, or Terminal Services Advanced Client. TSAC is based on the RDP 5.0, but comes in the form of an ActiveX® control. It has recently superseded the RDP client that ships with Windows 2000.

Another feature that can be compared between Citrix and the Microsoft product is load balancing. Essentially, if there is more than one thin client server catering to a community of users, a load balancing software piece ensures that few thin client servers would not be overwhelmed when the others would be under-utilized. Microsoft’s Windows Load Balancing Service (WLBS) is available for the entire Windows NT and Windows 2000 suite. It supports, of course, only Windows clients. Citrix’s Load Balancing option under MetaFrame is relatively more client-neutral. Servers that are load balanced with MetaFrame are available to all MetaFrame supported clients including:

- Windows Terminals
- DOS based PCs
- Windows (3.x, 95/98, NT) based PCs
- UNIX devices (7 versions of UNIX supported)
- Any device running a supported web browser
- Any other device running the ICA client



In addition, these clients can be connected via almost any type of connection (LAN, WAN, Internet, direct connect, dial-up, etc.) using virtually any protocol (TCP/IP, IPX/SPX, NetBEUI, PPP). Citrix Resource Management Services also provides extensive audit trail, comprehensive system monitoring, and the ability to construct detailed billing

reports. Load balancing is not a function typically used by small companies or small installations. In the same way, large companies often have more than one type of platform, often more than just Windows operating systems. Citrix does have a significant edge in this area. Moreover, Citrix’s load balancing system is a stable, well-established product with a proven track record. In the automotive world, one could compare the sophisticated, go-anywhere abilities of Citrix ICA to a Range Rover, which costs more than the average vehicle but can handle just about any road with aplomb and style. By contrast, Microsoft’s RDP application is much narrower in scope and ability. It could be compared to a Chevrolet Cavalier, which is more common, less expensive but designed for the more widely used roads. One thing is sure, more Cavaliers are sold than Range Rovers. Knowing this, it is no surprise that Microsoft’s RDP is quietly eating into the thin-client market, created initially in great part by Citrix.

The two main differences between Citrix and RDP thin client architectures are the protocols supported and the types of clients. Citrix is more flexible and versatile in terms of supporting a variety of clients and protocols. However, in the real world, IP and Windows constitute a vast majority of the market. This, combined with the cost factor does make Microsoft’s solution interesting to explore in Windows and TCP/IP only situations. **(The comparison at right tells the story.)**

In conclusion

Thin client architecture is not necessarily a magic bullet for all client/server communication bottlenecks. It can however be a valid option where the client application is Windows or UNIX-based and the number of clients are limited, remote, or have access to limited capacity lines. An iSeries Server can support multiple partitions, and is able to run NT in one of its partitions, providing the managing partition would run the native iSeries operating system. It is thus conceivable that a combination of a client and a server could run on the same physical machine, in different partitions, and



the signal to the remote (thin) clients could be distributed via Citrix or RDP actually running on an iSeries partition.

Using thin client architecture to distribute client signals over a private LAN or over the Internet minimizes the necessary bandwidth. All the heavy client-server conversations occur locally, within the same hardware footprint. Only compressed screen images. Keystrokes and mouse movements are transmitted to and from the thin clients. Choosing the right thin client product is a decision that can be affected by many factors. RDP and Citrix are both good products. If you are in the market for such a product, consider the long-term development of your systems. Changing architecture because of shortsighted planning may become an expensive operation. For more technical specifications on Citrix, see Appendix 1 of this article.

Platform Independent client-server architecture such as the one I described earlier is a strong trend amongst software vendors. No one wants to lose a sale “because the client did not have the right computer”. On top of this, increasingly, we are seeing remote and inaccessible areas being wired to head-office computers. The market for thin client computing has just begun to open up and already there is stiff competition to earn your

company’s hard earned dollars. Increasingly, our role, as information systems people, will not be to program every solution for our clients, but to find the right solutions, wherever they may be. Not only that, we will also have to use them in the best possible way. In this context, thin client architecture, Citrix or RDP, is just one more solution available in our toolbox.

With its ability to be partitioned, to handle multiple operating systems on one footprint, the AS/400 is now growing into its new identity (see Appendix 2): the iSeries. OS/400 V5R1 is now released. With this new operating system release, I can see the door opening for new and versatile iSeries business solutions using partitions in ways we have not seen yet, combining various software packages running on completely different operating systems in the same physical box. More than ever, the corporate IT marching orders are as follows: Squeeze the best computing value out of each specialized software package and out of each specialized box on the network. Internet Protocol, XML, thin client, what ever it takes, ensure all components work together. In this perspective, the iSeries’s unique ability to run several operating systems on the same footprint may be a huge competitive advantage in favor of this new way of computing.

In many ways, the AS/400 still suffers from the image of “old technology”. The common perception of the iSeries is that it is a new label on an old box. Until we start to fully use the iSeries’s unique partitioning abilities, the new name will indeed be just a new label. Our future as programmers using the iSeries server and indeed the future of this box will be determined by our ability to use the iSeries to its fullest ability. If we want to see iSeries gain that front-of-the-pack image, we former AS/400 programmers have to grow beyond our traditional OS/400 boundaries. High time to learn how to use Linux, Lotus Domino, Java, MS Windows UNIX and thin client computing. All these products can now run on the iSeries in a single footprint. What other platform can claim such characteristics? Who better than us to show the World?

Appendix 1: Technical facts on Citrix and Independent Computing Architecture (ICA) technology

Thinness on the Client: ICA requires only the equivalent of an Intel 386 processor and 640K of RAM to operate – dramatically thinner than X-terminals and PC alternatives.

Thinness on the Wire: ICA consumes, on average, 5KB of network bandwidth and is optimized for connections as low as 14.4KB. This allows it to operate predictably, even over dial-up and ISDN connections, regardless of bandwidth or application robustness. By contrast, the performance of “download and run” objects varies based on network bandwidth and object size.

Platform Independence: ICA is inherently platform-independent. Its modularity makes it easily adaptable to a variety of operating systems, including UNIX, Macintosh, Java and DOS, to deliver Windows, Java and UNIX applications to any client device, including the latest wireless and hand-held ICA-based devices.

Universal Client: ICA works with the latest Windows and UNIX applications, allowing applications to be developed with off-the-shelf tools and deployed with only one piece of ICA-based client software (which is less than 200KB in size).

Clients Supported	ICA	RDP 4	RDP 5
Windows 95, 98, NT, 2000	Yes	Yes	Yes
Windows for Workgroups 3.11	Yes	Yes	Yes
Windows 3.1	Yes		
DOS	Yes		
Windows CE	Yes	Yes	Yes
Macintosh	Yes		
UNIX Solaris, DEC, HP/UX, IBM, SGI, SCO	Yes		
LINUX (Red Hat, Caldera, SuSE, Slackware)	Yes		
Java JDK 1.0, JDK 1.1	Yes		
Browser Internet Explorer	Yes		Yes
Browser Netscape	Yes		
Direct Connect	Yes		
DP 5			
Protocols Supported	ICA	RDP 4	RDP 5
TCP/IP	Yes	Yes	Yes
IPX/SPX	Yes		
NetBEUI	Yes		

Protocol Independence: ICA is designed to run over industry standard network protocols including TCP/IP, Net-BEUI, NetBIOS and IPX/SPX and over industry standard communications such as PPP, ISDN, Frame Relay and ATM.

Future Citrix ICA Directions:


The solution extends the reach of Citrix server-based applications to the Nokia 9210 communicator

Appendix 2: Technical highlights on the iSeries V5R1

The following is a list of 3 items that in my opinion are breakthrough technologies that set the iSeries apart from the server crowd. This is by no means the full list of V5R1 new features but these are the ones that in my opinion will be the most popular. The three features listed below make it possible to have OS/400, Windows NT or 2000 and Linux all running on the same footprint. What other machine can claim this?

Logical Partitioning: Dynamic and shared logical partitioning will allow companies to manage multiple applications in a single server. Run up to four partitions on a uni-processor and up to 32 partitions on the largest iSeries 840 servers. A single processor can now be split into up to 4 partitions. On a multi-processor system, a portion of a processor can also be partitioned. Note also the dynamic nature of the partitions. The partitions can be modified without having to power down the system. For example, the number of processors can be increased to process nightly batch jobs.

Microsoft Windows Support on iSeries: The IBM iSeries allows you to deploy PC servers without adding the costs of managing a server farm. Through the Integrated xSeries Server for iSeries, you can run Microsoft ® Windows 2000 ® and Windows NT ® servers and your iSeries in a single system footprint.

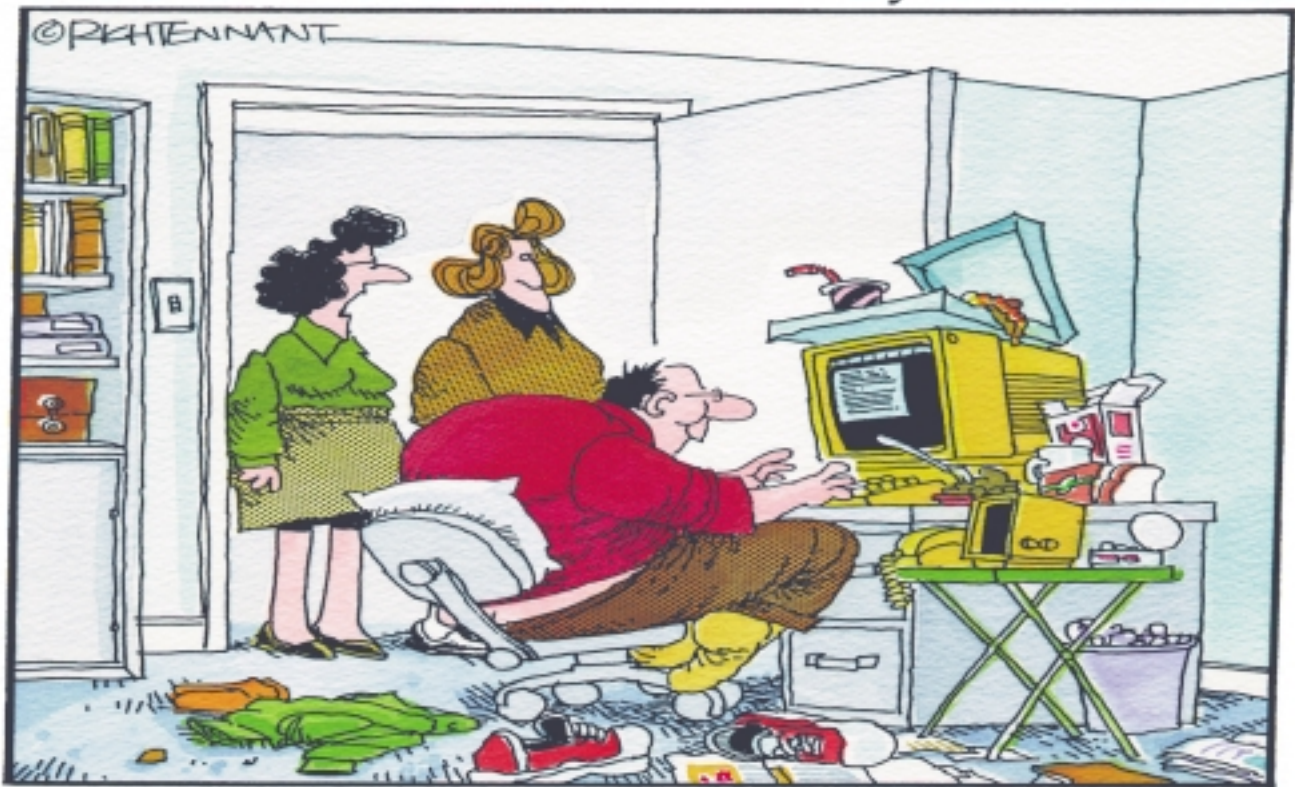
Linux Support on iSeries: The advantage of consolidating a number of Linux servers onto iSeries is that it makes them far easier to manage and more cost-effective. The individual Linux servers, each running in their own partition, are able to share processors, disk, tape, CD-ROM, DVD and LAN resources with the other applications running on the iSeries. With Logical Partitioning support, up to 31 separate Linux environments are supported on one iSeries server. 

Thibault Dambrine works as an independent contractor in Calgary, Alberta. You can visit the AS/400 related website he manages at <http://www.tylogix.com> or e-mail him directly at thibault.dambrine@tylogix.com.

The author disclaims all warranties, whether express or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Any trademarks and product or brand names referenced in this document are the property of their respective owners. Consult respective product manuals for complete trademark information. CanOilCo and PraccSoft are fictitious names.

The 5th Wave

By Rich Tennant



"SINCE WE GOT IT, HE HASN'T MOVED FROM THAT SPOT FOR ELEVEN STRAIGHT DAYS. ODDLY ENOUGH THEY CALL THIS 'GETTING UP AND RUNNING' ON THE INTERNET."

© The 5th Wave, www.the5thwave.com