

# RECORD OF SOCIETY OF ACTUARIES 1993 VOL. 19 NO. 4B

## LOCAL AREA NETWORKS (LANs)

Moderator: RANDALL A. KAYE  
Panelists: MARC BELEC\*  
GERALD H. PETERS  
W. STEVEN PRINCE  
Recorder: RANDALL A. KAYE

- Why are LANs growing in acceptance by organizations today?
- What do they allow you to do, and how is the business case built to justify them?
- Why are the human resources required to support LAN users always underestimated?
- There are many successes in migrating from the mainframe to a personal computer (PC)/LAN; what are the pitfalls to avoid?

MR. RANDALL A. KAYE: First, I'm going to have a few remarks on a personal level and then we're going to turn it over to our panel. We have three panelists, and I'll introduce each person in more detail as we get to him.

Steve Prince is with New York Life in Toronto, and he's going to talk from his perspective as the chief actuary of a life insurance company that has recently installed a LAN for the actuarial department and has taken the actuarial computing off of the mainframe.

Gerald Peters works for Trimark Technologies in Chicago, and he's going to explain how you really can have large mission-critical developments in a LAN environment, even though the platform has been evolving and developing over the last few years and continues to mature.

Finally, we come to Marc Belec from LOGISIL's Montreal office, and he's going to talk about what it takes to support a LAN. I believe many people are wondering just what is involved here and is it more than they expect.

I want to give you a few personal experiences from my background and a short history of my actuarial computing usage. Like a number of you, I started my actuarial career programming on mainframes, often in a time-sharing environment. Soon after I joined the company, we started to work with our first microcomputer. Some of you may recall that when the famous IBM PC was introduced in 1981, if you looked at the back of the IBM PC, there was actually a typical IBM four-digit model number, Model No. 5150.

Believe it or not, that was not IBM's first microcomputer. A few years before, IBM had, in succession, the 5100, 5110, and 5120. For me, the 5110 I used in 1979 was the first model that made sense to use for serious computing. It had a small, five-inch black-and-white screen over in the corner of the box, a nondetachable keyboard, 64K, and a manual switch to change between APL and Basic. It had two

\* Mr. Belec, not a member of the Society, is Senior Consultant of LOGISIL Consulting, Inc. in Montreal, Quebec.

## RECORD, VOLUME 19

eight-inch 1.2 meg floppy disk drives and a dot-matrix printer, and all for only \$30,000. Yes, those were 1979 dollars.

Even, as a small life insurance company, we were spending significant amounts on time-sharing costs for actuarial programming, so the cost/benefit analysis was really a no-brainer. It became obvious, since even we could save money, that time-sharing and mainframes would both diminish. I will say one thing about IBM, though. IBM's machines were rock solid; the systems on the IBM 5110 were still running ten years later.

When I first saw the Macintosh in 1984 and its graphical user interface, I realized that all those people who were taking computer literacy courses, who were worried about getting lost in the shuffle and thought they needed to learn programming in a hurry, were wasting their time. They didn't have to go to the computer; the computer would come to them.

Besides actuarial programming, I began using personal computers for word processing. It was a revelation, for I was now not putting up with the compromise of handing my work off to a secretary to type, proofreading it, and having it go back and forth, with me finally settling for something less than how I really wanted to express myself. I could now be absolutely certain I could say exactly what I wanted it to say. I'm not sure that it was better from a productivity perspective. I'm not sure that I spent less time creating my report, but certainly the quality was much higher.

You often see this with automation. People think they're going to save staff and be more productive. Instead, the quality of the work improves dramatically, as it does in the rest of the world, including your competition. Often you don't save on staffing.

As I was writing my reports, I found I would use an outline differently if I was using a word processor. Using a word processor changed the way I worked. It also ended up changing the way I think, probably for the better.

Later I started to work on a LAN. At first, it was only device sharing, since back in those days, printers and hard disks were so expensive. But that wasn't really a LAN.

Later, when real LANs were introduced, people would say connectivity is the "in" thing, and companies thought they needed a LAN. I have to admit that I was a stick in the mud. I said, "Why? Why do you really need a LAN?" Device sharing is not a good enough reason since the hardware and software are expensive, and you require a full time LAN administrator.

I would go to IBM seminars and the instructors would say, "Sneaker net is bad." Does everyone know what sneaker net is?

Sneaker net occurs when you take the diskette out of your machine, don your sneakers, and walk it over to the person you're going to share the data with. But back in those days I wondered how often we really shared data. Are we buying an expensive sledgehammer to get some extra little bit of productivity? I wasn't convinced.

## LOCAL AREA NETWORKS (LANs)

Finally colleagues said it was getting very difficult to manage all these PCs in the company. Employees were not conforming to standards, they were pirating software, and they were not doing their backups. And that's why we needed a LAN.

But isn't that really a management problem? You're trying to get your staff to conform to these methods and procedures, and you can't seem to get them to, so you're going to take this very expensive solution and hire a LAN administrator. Realize that you're admitting defeat from a management perspective; you can't solve this problem using normal techniques, so you're going to spend lots of money and build an empire to support the LAN. I still didn't buy it.

But now I'm no longer a stick in the mud. What converted me? The promise of work group computing. I've recognized this in my own business where staff in our Montreal office, experts in a particular technology, can write part of a report that I'm responsible for. They send it to me over E-mail and it works very well. E-mail is the beginning of work group computing. You can begin to see how synergies of people working together will evolve.

This, the beginning of work group computing, changes the way that I work, and it changes the way that I think, and try to solve problems. To me, that's the real promise of LANs. And seeing E-mail work for the first time is what turned on the light for me, and it might for you, too.

Of course, implementing a LAN can also be used as a catalyst for change. LANs will shake up your organization and change the way it works in ways that you don't expect; automation, like most technologies, will result in differences that you just don't expect.

I'd like to close with a story from an old IBM salesman. He said, "April 7, 1964 is ingrained in every IBM salesman's head." You can tell he's an old IBM salesman. April 7, 1964 was the date that the IBM 360 architecture was announced. He said, "At that time, IBM trotted out the visionaries and said soon there will be a terminal on every desk." Well, the visionaries are usually right. It takes a little longer to accomplish, and I don't think it turned out exactly the way the IBM visionaries expected.

Now I want to introduce Steve Prince. Steve is currently chief financial officer and appointed actuary of the Canadian operations of a New York mutual company. He has been there for five years. On the actuarial front, Steve is currently Chairperson of the Canadian Institute of Actuaries Committee on Life Insurance Practice. That committee's mandate is to provide guidance to appointed actuaries of life insurance companies on the professional performance of their duties. Steve spoke on compliance problems for foreign insurers in Canada.

Steve is also a member of the CIA's Committee on Continuing Education and is a past chairperson of the CIA's Committee on MBA Credits and a past member of the Education and Examination Committee. Steve was a member of the Society of Actuaries Task Force on Policy Illustrations. On the programming side, Steve's first summer job was writing actuarial pricing programs in Basic on a time-shared machine with 10(k) of RAM. He has been a hands-on programmer in (BASIC) Fortran (remember that), and APL. He has worked in APL in mainframe, PC, and older

## RECORD, VOLUME 19

desktop environments. In his current job, he hasn't done much programming in the last few years, but he is a hands-on power user of existing APL programs and PCs in general. He has a 486 laptop computer, which is frequently at his side when he travels.

MR. W. STEVEN PRINCE: One of the older desktop machines we looked at was not the IBM, but the HP9845, which we implemented in 1978, with the same sort of justification.

As you heard, my perspective comes from being part of senior management. I don't do much programming anymore. I like to push the technology to the limits, though. I use the LAN for word processing, spreadsheets, and we do our E-mail through our mainframe via the LAN, which seems like the way guaranteed to add to the overhead of doing so. We migrated off the mainframe to a LAN last fall. Randy was involved in the project, and he thought I'd be willing to share some of my observations with you.

The first question was, why did we decide that we wanted a LAN instead of one of the alternatives? The main problem with the mainframe was we were simply outgrowing the capacity. With all the actuarial forecasting, modeling, cash-flow testing, projections, value-added accounting, and so on, we just couldn't do it on the mainframe.

The way we could allocate our mainframe resource was to give the actuarial users a priority level, and this wasn't entirely satisfactory. If you gave actuaries a high priority level, they took precedence over the main administration system in the company, and all your service functions just ground to a halt. Conversely, if you gave actuaries lower priority, then nothing ever seemed to run, and you would have (literally) over-night batches not completing on time because they didn't get the cycles they needed; so we weren't happy with that.

Additionally, many of our actuaries had experience with PCs, not necessarily with LANs, but that was part of the learning curve. We hoped to gain more control over our operating environment. There was less risk of downtime, we supposed, and I'll get to that. It should be easier to shift out of or between our APL programs, spreadsheets and word processing if it's all in the same environment. Additionally, we were using an IBM version of the APL language on the mainframe, which wasn't entirely satisfactory either. There were ways to upgrade that, but again, the cost was prohibitive and so something had to give.

In terms of our business case to justify the LAN, luckily for us it was mainly made by the information systems (IS) department. Those people wanted to get us off the mainframe, please. They would have loved to have upgraded the mainframe, but you're looking at chunks of \$1-2 million to make any kind of incremental mainframe adjustment, and that kind of money was prohibitive. Our IS department was in favor of LANs. It had other applications in mind for LANs. It was looking for any other department that was willing to use the LAN to get started on it, and that's fine. Many actuaries already had PCs, so we figured the LAN was going to be little incremental cost, simply the cost of the LAN rather than all new PCs. In fact, we wound up getting all new PCs anyway to handle the extra power requirements.

## LOCAL AREA NETWORKS (LANs)

Why do LANs offer some other solutions? We looked at stand-alone PCs, and I have used sneaker net in the past. It worked in some environments. In the larger environment, the experience is that people just don't back up their stuff, and that was potentially serious with things like valuation work and programming modifications that one person makes and doesn't document or doesn't back up. We looked at RISC/Unix technology, which is supposed to be the wave of the future, or we have been told was the wave of the future. There is certainly more power available in a RISC machine than on a PC, but it wasn't enough.

If the question was should we get a bunch of RISC machines, one per desk instead of one PC per desk, yes, we would have gotten more power; but that cost would have been literally three or four times what we had planned to spend. We considered getting one large RISC server for several clients or for all the actuaries, and we were worried about having enough power. What we eventually decided to do to handle our monstrous month-end and year-end valuation programs was to just buy a couple of extra PCs, 486 machines, and leave them on someone's desk. You turn them on and let them run. Then the person also has his or her other PC for regular day-to-day work. The relative power of the RISC machine, we decided, was about one and a half to two times as fast as a good 486 machine. That didn't justify using a central server for the valuation, compared to having several PCs running in parallel.

Additionally, we were also going to need word processing and spreadsheets, and you can do this through a RISC/Unix machine in a DOS window. I wasn't convinced with the argument that I should buy a RISC/Unix machine because it has a DOS window. Well, why don't I buy the DOS machine? Mainframe upgrades were not an option due to the high cost.

We looked at converting our existing APL programs into some more efficient, probably batch language, and we have over 20,000 lines of existing APL code, which we were not in a hurry to rewrite. Anyone who has worked in APL would know that 20,000 lines of APL is literally 200,000 lines of many other languages.

Actuarial programming is dynamic. Even if we had converted all this stuff at least once, you're then stuck with the ongoing changes in regulatory evolution. We couldn't afford to be locked into a rigid system, and I'll expand on that in a second. Two other factors in favor of the PC/LAN were that the PCs are still growing in power. A few people said, "Well, what about outgrowing 486 machines?" Well, that was a concern but, heck, the 586 machine was in the wings and has since come out. The 686 I guess is being worked on. The clock rate of 486 machines has doubled, and we weren't worried that our requirements were going to grow any faster than the technology was growing.

Another attraction of LANs was that it allowed the easy sharing of occasional use software, such as *Harvard Graphics*, which several people will use once in awhile, but we didn't want to buy a copy of that for everybody.

Why do we do our own actuarial programming? This was a significant consideration in deciding to go with the LAN environment. As I said, the regulatory environment is always changing. We decided that we can't live with someone else's programming.

The flexibility and adaptability of the programming to the environment would have been the vendor's problem, but I wasn't convinced that would happen fast enough or reliably enough for us to bet the company on it, so to speak. Speed was an issue. Customization is an issue. Any good vendor software for actuarial work will let you handle all manner of unique product features, but the ones we looked at required you to set up tables and tables of numbers; for instance, your unique cash value scale or your unique table of mortality charges in a universal life product.

We would have been in the situation of writing programs to create the tables to feed into the programs. With our in-house models, you simply set up an APL function or subroutine that creates the table, feeds it into your pricing program, and does whatever analysis you want. You can even have the program automated to iterate on the charges to get the desired profit level. We didn't think that capability would be available with vendor software. It was certainly not available with vendor programs we had looked at.

Reliability was a concern. I've never heard of a problem with a vendor's actuarial program, but one problem with using someone else's work is the people running the programs tend to take everything the program produces at face value. Well, the program says it was \$1.75. Who am I to argue? Well, there could be several reasons why the \$1.75 isn't correct.

By using our own programs, I think this forces us to maintain a healthy level of skepticism of any result. It might have been a programming error. It might have been a data error. It might have been a misunderstanding of what the question was, and so on. Everyone is just a little skeptical of any answer from our own programs, and we just have that extra measure of checking or verifying: so-and-so came up with \$1.50; maybe \$1.74 isn't so bad.

Finally, consider functionality. We have, frankly, a nice actuarial package. It's been developed in house back in the days when I was programming. Frankly, I have written more than half of it, although with each passing year more of it gets thrown out and replaced with something new and different. It integrates our pricing, forecasting, and valuation into one package. Quite literally, when we do our pricing work, the valuation basis we factor into our pricing is the actual valuation basis we're going to use. It's also both the pricing and valuation basis we use for forecasting, so everything is always consistent because it's built in.

We were not able to find a vendor who could do all that. Some had excellent pricing programs that didn't attempt to do modeling or valuation or asset valuation programs. Some didn't pretend to price, etc., and so we would have been faced with two or three sets of software, none of which spoke to each other very well, and all of which required additional programs to be written to create tables to feed into them in the first place. All things considered, I decided we would stay with actuarial modeling in house.

Why are human resources always underestimated? For this, I am anxious to hear what the rest of the panel says because I don't understand it myself. We have actuaries who seem to be able to run PCs on their own. They learn word processing, and a good actuary or a good, experienced user becomes more proficient at his or her

## LOCAL AREA NETWORKS (LANs)

word processing package or spreadsheet or whatever than any technical support person, because the technical person by definition is not a user.

The result, is if you really get hung up on your application, certainly for an APL program of an actuarial problem, you cannot go to your technical support people for help with that, and so they aren't much help to you in that type of thing. In terms of the things where they can help you – while your LAN is frozen, your server is down, or your printer isn't communicating with something – yes, we go to them; but I don't know what the problem is in the first place, so I never understand why it's not working when I want it, and I'm sure these fellows will have answers to that and other burning questions. My conclusion as a user, and these fellows will speak to it, is, it's always frustrating to have to look to your technical people for support because you never understood what the problem was and they never fix it fast enough. We'll hear about that later.

There are pitfalls to avoid. Our transition went remarkably smoothly because the mainframe APL model we were using had been migrated up to the mainframe from stand-alone PCs five years earlier. The architecture of the programs was, if I may say so, very well designed. For instance, all of the printing that was done anywhere in the system was done through a user function called PRINT. We did that on purpose at the time so you could make a duplicate file of all your output on a hard disk. Having built that in as a requirement, you now had routing capability.

You simply had a menu that when called up, it said, "Where did you want this printed? (1) on the printer; (2) on a file; (3) on the screen; and (4) any combination." Once all your printing is done through that function, the fact that you're now in a different environment and your print commands are different or your routing is different, you simply change the one print function, and instantly everything is fixed.

Our programs were menu driven, and we'd simply save the old menu driven programs from the PC versions. We update the menu content, but the programs were easy enough to change. Because it was so simple, we converted the main application program to the PC and did our bench marking about whether or not to go with the PC or RISC using our actual main live application. When we reached the decision of where to go, it was almost a no-brainer to simply convert a few other programs to the PC environment.

How is it going? Everyone is still learning the LAN. Our technical people are expanding the LAN. We had several separate LANs in the company previously. They're now moving to something called a super server, which is going to coordinate all the LANs in the company in one monstrous package, which frankly still freezes up occasionally and these fellows will explain why.

*Were the PCs the right answer rather than RISC machines?* It's certainly adequate for our requirements. We'd be happy if the PCs were twice as fast as they are. In six months they will be twice as fast as they are. Generally, we're looking forward to ever more powerful PCs, but our set-up now seems to be meeting our requirements. Those are my comments as a user.

## RECORD, VOLUME 19

MR. KAYE: Gerald started his career as an actuary for the Kemper Life Insurance Companies and soon became involved in the data processing activities. He managed Kemper's conversion to consolidated functions ordinary (CFO) and later conversion to the Cybertek Systems. In addition, he managed all of the administrative departments. He was also instrumental in the implementation of an industry-first team processing concept for new business. Later, he worked for Cybertek as vice-president and directed all software marketing activities, project installations, and data center operations for the company's largest region.

More recently, Gerald has held senior management positions with Kemper Financial Services and Keystone Provident Life. He has directed both data processing and administrative activities and has managed the installation of several major new software systems. Currently, he is chairman of the board and chief executive officer of Trimark Technologies, Incorporated, an innovative new life insurance software company. Trimark specializes in the development of PC, network-based policy administration systems. Gerald graduated from Drake University in 1967 with a BS degree in actuarial science. He is an Associate in the Society of Actuaries and a Fellow of the Life Management Institute.

MR. GERALD H. PETERS: What I want to talk about is a little success story we have had in the development of a system we call Lanmark. The LAN in Lanmark stands for life administration network, not local area network.

Before we begin, just to give you an idea of what I'd like to cover, I think you need to understand what kind of time frames it took to get to where we are, why we set out to do this, what we thought we were going to accomplish, and how we did it.

We need to talk about the platform (because the platform is a critical element here), some of the challenges we ran into, and obviously most important, the results.

Trimark started in 1988, and we entered into an agreement with one life insurance company to explore the possibilities of developing a complete life insurance administration system in a network environment. We signed an agreement in September 1988. We delivered, installed, and put into production agency administration functions in March 1989, only six months after we had started. A year later, we converted the first block of policies. Two years after we started this project, we had converted a whole product line: 150,000 policies which, by any previous standards, was a very fast project.

Why did we decide to do this? As you can tell from my background, I've been involved in systems work since the beginning of time, and I wrote my first programs in the late 1960s in report program generator (RPG), and have been involved in systems conversions, it seems like now, forever, and never really in a satisfactory environment.

First of all, the cost was always an issue. Anything you did in a mainframe environment would cost tons of money. The time to get things done was too much, and there were endless priority lists. When do you need that? Oh, I need it tomorrow. When will I get it? Oh, a year from tomorrow. Not acceptable. Also, the users never got the functionality they really wanted. The things on the priority lists were

## LOCAL AREA NETWORKS (LANs)

always those must-do things, not the things that really helped productivity or anything else.

The vision was aggressive (when you consider this is over five years ago): to eliminate the systems constraints involved in bringing new products to the market. In other words, our goal was to eliminate the systems arena in the critical path of bringing a product to market. Our goal was to let the marketers, the actuaries, and the legal folks worry about those issues. When they decided what kind of product they needed, it was not going to be a systems problem. We also wanted to administer these complex life products that we have in a single system, all with an integrated design. Now, that's been something that the vendors of the past have talked about for years, and no one was really ever able to bring to the table.

We set out to build a better mousetrap. How did we do it? First, we started with a new platform, which I will spend some time talking about. The major reason that we looked at a new platform is the same reason Steve talked about. The mainframes, quite honestly, were not powerful enough to do the job that needed to be done.

We started with a small group of experts, people who had a lot of knowledge in the various systems out there and a lot of life insurance knowledge. By doing that and by providing a platform that was much more productive, we could do the job with a lot less people. That eliminated all the bureaucracy issues, and it eliminated a lot of the communication issues in terms of getting things done.

We looked at new solutions to old problems. As I have said, we had no albatross around our neck. We had no restraints. We could do anything we wanted. We had no system we had to build off of. We had no limitations in that regard.

One of the interesting positives, I think, of this whole project was short-term accountability. This agreement we signed with the insurance company we were doing business with had a 24-hour cancellation clause, which is sort of unheard of in the world of software development. As a result of that, it really focused our attention on getting things done and getting things done quickly.

The platform? LAN only, by design. We wanted to take advantage of the powers of a LAN in terms of really distributing the processing power back to the end user. In mainframes, everything was centralized, and the LAN offered us this opportunity to get back to where we've always wanted to be in terms of a distributed processing environment.

Using PCs only gives you unlimited power. Every user you add to a network adds power to the network, because you add another PC. In the mainframe environment, you have a fixed amount of power. You add a user to the network, and you, in fact, cut the resource to every other user. Another reason for using PCs in LANs is there's a lot of packaged software available that we could use in this process, such as word processing systems, and so on.

We decided to use a relational database. Now, that's been a hot topic in more recent mainframe development, and the reasons are obvious. It's a lot easier to add functionality in this kind of environment, because if you need more data, you can add

a new file and define the new data you need. You also have the ability to get at data a number of different ways by using a relational database.

We chose to use a fourth generation language (4GL) for the development of this system. The language is called Magic. It's really not a language per se. It allowed very rapid development, and because it's a 4GL, it allows immediate execution of the programs. So there's no compiling, no job control language (JCL), and all those kinds of things you have with a mainframe. Again, with the power of the PC, there's plenty of power there to interpret the 4GL code as it's executed.

This all sounds well and good. There were a few challenges along the way, and there continue to be a few challenges. One thing we determined in the course of five years in continuing the development of this system and its functionality is that all great solutions are simple and elegant. As soon as any given function or feature becomes complex, it's not going to work well because it's going to be too hard to understand and too hard to maintain.

As we were looking at the various insurance functions and the various product features that had to be administered, we often, what I call, turned the table upside down; because the obvious solution was to look at it like we always did in the old days. Many times as we focused on this simple and elegant concept, things would look very complex and we'd get into some heated discussions about how we should do these things. If after an hour or so we couldn't reach a simple and elegant solution, we'd stop and then reconvene the next day or the day after. By rehashing it and looking at the subject in different ways, we were able to come up with better solutions.

Obviously, as you look at different ways of doing things, getting away from the old mentality, we have this problem of getting people accustomed to thinking about processing a new way. Not only is the network new, but a lot of the things that we did and how we do them are very, very new. Not only for our own people, but also for our users, our customers, this teaching old dogs new tricks is a continuing major issue. As I alluded to before, we actively encourage disagreement in the development process, because by arguing about the various ways you can do things, generally some new and better idea will result.

Everybody says that networks are fine, but can you really handle the volumes? What if I have 100,000 policies, 500,000 policies, or a million policies? There are performance issues here, there is no doubt about it. It's not as simple as saying it's no problem. Obviously when you design a system for a network versus a mainframe, there are some things you have to do differently. The nightly cycle concept has to go away, which we did eliminate. For many other things we took very different approaches to how to do them.

One of the great things about the old world of mainframes was that everything you had on a policy, for instance, was in one record and you could get to it very fast and process it very quickly. The other side of it, with the relational database, is you end up generating a lot of data. The more data you collect, and sometimes you need a lot of data at the work station to do what you need to do, the more time it takes to get it there.

## LOCAL AREA NETWORKS (LANs)

In order to overcome some of those issues, we've invented a few things; we've written our own software to do more than these two things, but these are critical:

- Work station caching is where the read-only data (things like programs, tables, rate files, and so on), are read into the work station and kept throughout the day in memory so that if the work station has to access those data again, it just plucks them out of memory. That we have found has cut network traffic in a ratio of three to one.
- Transaction basically means that, as you do work on your PC or activities are processed, all the data that are updated or created are maintained in memory until you are done with that transaction, and then the data are written out to the database. Again, we have found that not only does this improve the whole reliability issue in terms of not having things half-done if a PC breaks or somebody pulls the plug or shuts it off, but also, again, it cuts network traffic in a ratio of three to one.

We were doing things to overcome the traditional concerns about network environments in that regard.

Obviously we have seen and continue to see the hardware capacities increase. When we started the company, we had 286s and we were using 286 servers. We said, "What's going to happen when we run out of capacity?" We said, "Well, we don't know the answer." Fortunately, new high-speed servers have come along, the 486 PCs, and we continue to see the improvement in that whole process.

One thing about a very productive environment, one where you can develop a lot of functionality – and I will contend that probably Lanmark has in it more functionality than any other life insurance administration system available – is that testing becomes a major problem.

Obviously we have an integrated system. That means there isn't a separate billing system, there isn't a separate commission system, and there isn't a separate issue system from the in-force system. It's all an integrated platform. That means, if you make a change in one arena, it may be quite innocent, but you may change something going on somewhere else. That adds to the testing process.

The traditional way of looking at testing has to be reviewed. If you have people sitting down every time you make a change and going through all the traditional regression testing, it takes far too long. As a result of that, we are, as we speak right now, in the process of developing a mechanized means of testing. Now, I know many people have talked about that in the past and there are some implementations of it, but we really believe that you have to be able to compare a production operation for a period of time using the production version of the software versus the new version of the software and make sure that the data are not changing other than you expected.

What are the results? Well, we have a system, we think, that has unequalled support. We support, as we say, one product, that being life insurance, all traditional, universal life (UL), interest-sensitive, variable products, including health, and we handle all administrative functions. In the course of doing this, we've designed some things

that make the implementation of new products a lot easier – something called profiling. Products are made up of a series of features. Is it funded or is it not? What kind of funds are used? Do you bill it or don't you? Do you allow withdrawals or don't you? If you have withdrawals, do you have a free withdrawal?

Once we developed a product feature and programmed it, we made it available to any product by the way of profiles. Basically, it's just like a light switch. If you wanted to define a product, if you want to turn on a feature, you flip the switch and use that feature for this product. We've done the same thing in terms of functional capabilities – something we call functional commonality. If there's a function that needs to be performed in one area of a life insurance company, we do it the same way no matter what other areas might use the same functions. Again, unless there's some new feature that we don't have in the system, you can put new products up very quickly by just turning on or off profiles.

Consider the results in terms of savings. We have saved our customers a ton of money. There have been various quotes from customers talking about millions of dollars. Obviously if we can put new products up that quickly, it does save a lot of money. We also have saved a lot of time for our customers. One of the real things that people talk about is the ease of getting new products on the system, and that's what we set out to do.

We believe we've eliminated the little marketing guy being frustrated with his sales going down the tubes. We believe we've found a better solution to the old problem.

MR. KAYE: Marc Belec is our next speaker. Marc is a senior consultant at LOGISIL's Montreal Office. He is our resident LAN guru, and he is not a member of the Society. For the past ten years, Marc has specialized in the use of microcomputers, first at a large distribution chain, supervising the increase from ten pre-XT PCs to over 900 networked with gateways to minis and mainframes.

He has also implemented portfolio management connections between Zurich, London, Montreal, and Toronto for a Canadian banking and financial services conglomerate. More recently, he has formulated the PC/LAN and office automation strategy for a major Canadian life insurer and early in 1993 Marc presented a LAN seminar at the LOMA Systems Forum in Dallas.

MR. MARC BELEC: When this presentation was given at the LOMA Systems Forum, it was more oriented towards technical people, so I'll try to give it a new slant and move away from the techy side and explain what you should be asking your information systems department for in terms of support.

You'll have to excuse my accent. I am a French-Canadian, and I was told that my English speaking is better than your French understanding!

We'll talk about what you need to know to have effective support. The knowledge of your environment is critical. You need to know what's installed, and your IS department needs this information. What kind of hardware requirements do your packages need? We'll talk briefly about keeping all this inventory information up-to-date.

## LOCAL AREA NETWORKS (LANs)

What kind of good support people do you need for your organization? Do you need "propeller heads," or do you need people who are more oriented toward your business? We'll talk about some pros and cons of both sides. What kind of training do these people need to offer better support? There are many technical certification programs out there. How valuable are they? How can you best deploy your resources to get the utmost support? Steve is concerned that there are never enough people around. We'll see how we can better address that problem.

We'll talk briefly about strategic alliances, how your IS people can get some vendor help. Usually there's a lot of free help around, if you know where to get it. I won't talk a lot about the tool kits, which I refer to as an arsenal here, because that's a little bit more technical.

We'll talk about your intelligence network. Basically, it's keeping track of all the problems you've had and trying not to reinvent the wheel every time you have a question.

We'll also talk about a defense strategy, basically a contingency plan, identifying key systems, making sure you have backup procedures, a certain amount of redundancy throughout your network, and how to preserve the peace and keep everybody happy. Basically, this is a more proactive approach to support.

At the end of this session you should have an idea of what the major components in your system are, how to best protect them, and how to implement a cost-effective support strategy utilizing the best resources where you need them.

What do you need to know about your software, and why do you need to know it? First of all, you need to know which software is installed for upgrade programs, making sure you're always up to the latest version. That's generally a good idea. Of course, everyone here wants to be legal, so you need to know how many licenses you have.

You need to identify your mission-critical applications. If you're building a system, you want your IS department to be aware that this system cannot go down.

Who are the users on this system? Usually, they have to have access to it full time, so you need a contingency plan. If there are upgrades, you need proper user training.

Where did this software come from? Is it off a bulletin board? Is it a potential virus access point? What revisions are available?

On the hardware side, you need to have a hardware inventory for managing service and warranties. That's a big issue these days. Service contracts are very costly and hardware costs have gone down drastically. Something you might want to consider is to just have backup machines. Having them might be more cost effective than having a service contract on all your equipment. You can have some technical people in-house just to replace the machines. And when older machines break down, you might not even want to have them repaired anyway.

Of course, you need to know what hardware is installed for internal corporate purposes such as amortization and recycling strategy. You need to identify your potential failure points for contingency reasons. You need to know the product characteristics and the physical connections, because, on a network, communication servers and gateways to mainframes are components that are usually critical. And they are quite often underdocumented.

Good support people are business-oriented. It is a lot easier to train someone in the technical computing aspects than on the business aspects of your operations. Explaining the applications and which ones are critical for your operations is more difficult than just fixing hardware. As much as possible, try to recruit people-oriented support staff, not just some techy who doesn't explain what he or she is doing to solve the problem; this usually generates frustration.

Try to find someone inhouse who has expressed an interest in computers, who already knows your business, and then train that person as a network administrator with corporate backing for higher level functions.

External resources also can be helpful, such as consultants. Outsourcing is growing and has been found to be cost-effective, especially in providing localized support over a wide area.

There are several ways to train support people. Both Novell and IBM have vendor certification programs. If you're looking for support people and they say they have been certified, all it means is that they've been able to study and pass a test, which is good in some respects but not when you're looking for technical people. Go with experience. People with no experience but who have attended a certification program won't be able to properly diagnose problems with your system.

Formal training is good, but it is usually costly. And you can often try to get free seminars from suppliers. Many vendors will provide a lot of information over a lunch.

To address Steve's problem, one of the best ways I've seen is to develop localized support people. These could be high level users who know the application and who will help the branch office handle its regular day-to-day problems if it needs help, whether it be with a technical aspect or with the application. Then the branch can call a second level of support, which is usually at the corporate level.

Also, local external resources, such as vendors or consultants can help out for certain problems. Local support is critical because otherwise you'll have a high level of frustration in your user community if something goes down, and users always have to rely on the head office or a centralized support scheme.

Strategic alliances are important. Users need to have good communications with their main internal resource, usually the IS department. The IS department must be aware of your application and your requirements, as well as external resources to be called upon, so as to know of new products or enhancements that can be used in your business.

## LOCAL AREA NETWORKS (LANs)

Let's discuss tool kits. There are many products that will allow a centralized support group to offer support to branches remotely. This remote-control software allows a technician to take over a local PC and perform his diagnostics.

Encourage your support team to log all service calls. Information such as what kind of problem and where it occurred will point to additional needs. Do you need to do more training? Do you need to upgrade your hardware? What is the general cause? A lot of companies reinvent the wheel regularly because they don't keep track of what's happening on their network. And you may even be able to justify additional support staff.

The users must also support the effort. On-line help and the use of E-mail will cut down on support costs. A simple screen can be created to allow the users to log in problems and send them to the IS department. Of course, I've already mentioned a little bit of remote-control software to have access to remote sites to do easy support.

As for the defense strategy, all your critical equipment must be in a secure environment, with uninterruptable power supplies (UPSs). Common sense in the mainframe environment is quite often overlooked when you talk about PCs. You must keep the same kind of rigorous environment and rigorous security.

As for software security, of course you need virus protection. I've attended many seminars and, on a show of hands, at least 60% of the audience had been infected by a virus at one point or another. They're out there, just waiting for an unprotected PC.

If you have remote sites, you must have control of the access points. If you have any kind of critical or confidential information, you must keep a record of anyone who accesses your information from a remote site.

In order to preserve the peace, you need to plan future capacity, to remove bottle-necks on the network, such as inadequate band width for the network traffic.

Try to be as up-to-date as possible on software. At some point, vendors simply stop supporting old versions. I wouldn't always go to the next available version. Leave it to the innovative people for a couple of months, at least to make sure that all the bugs are ironed out, but I would move up systematically. And make sure that your user and support people are properly trained on each upgrade.

As to hardware, if you want to make sure that your application works everywhere, try to find the "lowest common denominator" PC that you have in your company. If you have an old XT somewhere, try to run your application on it so you at least know what platforms it will function on and what kind of performance you can expect. This should cut down on support calls because you will know how much performance you can get out of that machine. Finally, make sure you have a contingency plan for all your critical applications.

MR. SAMUEL B. VENABLE: This is a question for Marc. I didn't catch who is to keep track of the software documentation. It's a problem we seem to have at our company. We have close to two dozen LANs and the IS department takes care of

the hardware side and the basic standard configuration; but then when it comes to the software documentation and license tracking, IS purchases the software for us and all that we get really is just purchase orders showing file copies of Lotus such-and-such purchased. I'm wondering how best could that be tracked. Sometimes I have one license to purchase with a purchase order under one cost area, but the cost gets split up into other cost areas; there are just a host of problems that occur here.

MR. BELEC: There are many good software tools that will allow you to keep track of your software packages. Some allow you to manually keep things up-to-date, like Computer Associates Netman. Whenever you purchase software, it will attribute it to a cost center. Even if you have one purchase order with five licenses, you can break them down. There are also some network-based systems such as Network Examine, that will automatically keep track of the log-on time for each user, and you can actually bill the cost centers per use of the license, just as on the mainframe.

If you're also talking about documentation, the developers are responsible for maintaining documentation, but they should, of course, give a copy and proper training to the support staff in the IS department. Does that answer it? Because I think there were two parts to your question. Did I address it properly or not?

MR. VENABLE: That addresses most of it. I guess my concern is, if someone were to come in and audit, what would we need to show to prove that we did actually purchase all the copies of software that are in use? Because as things get upgraded, we don't always get the user documentation with the use of LAN special licenses.

MR. BELEC: Especially right now. Both WordPerfect and Lotus, if you're a big enough company, have stopped sending you documentation. You can just purchase a piece of paper that will tell you that you are entitled to 200 licenses of their software. Novell network software includes an accounting function with which you can track how many simultaneous users have been using Lotus. With that tool, you can justify purchasing more when you've reached, let's say, 205 simultaneous users. When you buy additional copies to remain legal, but that piece of paper should be able to prove how many simultaneous users of the software you have at one specific moment in time.

FROM THE FLOOR: This question is primarily for Randy, but anybody may want to take a shot at it. I agree with your observation that what's going to really drive LANs is groups of people working together. I think even in the industry now more and more departments are forced to work on projects in common and a very easy way of doing it is each person gets his or her assignment and goes off and creates a little file and you put the pieces back together at the end.

One of the problems is that individual users still tend to work as individual users. When you get the pieces to put back together again, you get files that are three or four times as big as you would expect and probably 30 or 40 times more complicated than you would expect. It puts some strains on your hardware, and maybe just the program doesn't run very well. In terms of manufacturers of software and hardware, are they thinking along these lines? Are they trying to come up with some features that would take this problem into account, or have they done so already?

## LOCAL AREA NETWORKS (LANs)

MR. KAYE: I haven't seen anything along those lines, but I would say that not only does it put a strain on hardware, but also it puts a strain on your personal relationships with your coworkers when you say, "Thanks, that was great. That was much more than I expected," and then you edit it ruthlessly, to what you really wanted. I've had that experience, too.

MR. CHARLES S. LINN: You mentioned that you were in the process of going to or have gone to a super server and combining multiple servers. I was wondering if you could elaborate on what you see as the advantages and disadvantages of that. Is there any impact to the end user? Is it much masked behind the server already being used?

MR. PRINCE: One advantage is that you can now share data with users on other floors. When we were working our way into this, we had one LAN per floor. In my own case, my department is spread over a couple of floors, and so, even though there were LANs, it was hard to get stuff from one LAN user to the other. In terms of impact on users, there's a teething problem. I don't mean to suggest we don't get things working eventually. Once it's working, it is transparent to the user. You simply log onto a LAN, and from the user's perspective, it looks the same as the old LAN; but now other people can share data with you.

MR. JEFFREY T. ROBINSON: I've always found that with pluses there are minuses. Nobody spoke to the question of the disadvantages of a LAN. Steve, were there any? I was involved in one, and I knew it degraded the system. The reaction time or response time went way down. Could anybody answer that question? What do you have to watch out for? What is not considered in these things? It sounds great, but there must be a downside to it.

MR. BELEC: One of the major downsides is the distribution of software when you get into graphical user interfaces (GUIs), such as Windows. When you have graphical information traveling through your network, you're getting a lot of bytes moving through the wire, and that can be a major performance hit. You need to plan very carefully. You may find that you want to go back to the old scenario and encourage users to have the applications and information at each individual's PC, and transfer it to the network on a periodical basis.

MR. PRINCE: There are some negatives. When we decided that the actuaries would go on the LAN, we analyzed the printing requirements. We got the right printer to suit our needs and that worked well. Now that we're on a LAN, other people are accessing our printer when theirs is busy, and now our stuff isn't getting done.

We're starting to have the sorts of problems you had on the mainframe. Security access is one. People are only allowed to have certain data, and you're at the mercy of the system administrator as to whether he or she correctly anticipated which data you wanted to get at.

In my own case, the administrator forgot that one of the departments works for me and I want the department's data and I can't get at the data, even though it's all on the same LAN; so I need to get access to what I used to have access to. I don't see a huge difference in the LAN. As I said, one of the attractions of LAN over PC was

RECORD, VOLUME 19

the data backup, so I personally have a slightly different data backup procedure than I used to.

MR. ROBINSON: That leads me to another question. What does a LAN administrator do? One problem with the PCs over mainframe shop was lack of discipline. You mentioned backups and documentation. Is that what the LAN administrator does?

MR. PRINCE: Honestly, and I'm not kidding, I don't know what a LAN administrator does.

MR. ROBINSON: Do you pay that person a lot of money?

MR. PRINCE: We pay the administrator a fair chunk of change. While the administrator was setting up the LAN, while we were moving to a super server, and so on, you sort of had a sense that the LAN administrator was working on this. In what should be a stable environment, I don't know what the ongoing job is.

MR. ROBINSON: I've heard that you need one, but why and what does the person do? Can somebody answer that?

MR. BELEC: Basically the position generates my reason for being.

MR. KAYE: And the other reason?

MR. BELEC: The other reason is there is a lot of administration in user creation, user rights management, making sure the backup procedures have gone properly, adding additional people to the network, both physically and logically, hooking up PCs, hooking up printers, and, every time you install a new piece of software, making sure it will run properly in your environment. That can be a lot harder than it seems. Upgrading your network operating system is not something that you want to do lightly either, and trying to tweak the system to get the maximum performance out of it. All these things together don't sound like much, but I can guarantee you that the person is usually occupied full time.

MR. ROBINSON: That's my next question. Is it a full-time position or is it easier to take one of the users like they do on a Xerox machine, either a senior program men or a senior operator?

MR. BELEC: Let's say you have a branch office. You can have a well-trained power user do local support and user creation and rights management, but you need at least one technical guru somewhere or access to one. It could be outsourced. That can help you in a crunch. If you have a hard drive that fails, what do you do on your network to make sure that the proper redundancy is built into your system? Certainly you need someone at set-up time, but the technology changes so quickly that there's always something new coming on. That's why you need at least one highly technical person available.

MR. ROBINSON: That's my next question. Do you need two in case one is on vacation?

## LOCAL AREA NETWORKS (LANs)

MR. BELEC: Some people work that way. For some of our clients, we are the second source of people available to them. When they're on vacation, the consulting firm can provide a backup person. Consultants usually have the experience, and they can also provide you with a new perspective on certain problems that might not have been seen by your local person. For short periods of time, I believe that's the best alternative.

MR. ROBINSON: I have one more question for Gerald. You talked about a testing solution. Can you give us anything more about that?

MR. PETERS: What we're in the process of doing is putting together an environment where you can actually parallel the production environment for a period of time, taking the output of the production environment, and comparing the data against running that same production activity against the new version of the software. Obviously, you have to have another server available. You have to have these things connected. Some other features we have available in our software are duplex server capability, and so on.

My whole point here, having installed systems and having lived through the Trimark experience, is that the traditional way of testing software as we go forward has to change because you just don't have enough people to do it right. Testing has to be done by the people who know what's going on, but they test what they think they need to test. Then you put a version in production, and suddenly this happened or that happened. Well, we didn't test that. I think the long-term issue is that we have to continue to do a lot of development. A lot of functionality has to continually be added, and we have to find a mechanical way to do the testing. We think we know of a solution. We're in the process of developing it, and we'll probably have it delivered by the first quarter of next year.

MR. ROBINSON: See the next issue of *CompAct* (formerly *Digital Doings*) the newsletter of the Computer Science Section. I've written an article on testing, and I agree with you. To do the things that have to be done, particularly in administration systems, or any system, you have to test and that really isn't done well now by most people.

MR. PETERS: Right, I agree. The more functionality you have and the more functionality you continue to provide, the more critical the function. If you have a static system that doesn't change very much, this issue is much less a concern.

MR. KAYE: Before we go to the next question, I have a comment also about what a LAN administrator does all day. Well, in many respects, the administrator does just what a mainframe IS department does all day, too. Even though it's a LAN, it doesn't mean that you don't have to do those things, because, yes, they are mission-critical applications and, yes, it's a corporate resource and, yes, it's very important. The difficulty, though, is that it's still a maturing technology and everything is not in place. There are different attitudes that apply. For example, I can find mainframe people immediately because they believe in preventive maintenance. A PC administrator believes in redundancy.

RECORD, VOLUME 19

MR. A. STEWART WILDER: My question for any of you gentlemen concerns user inertia. It's one thing to migrate from the mainframe environment to the LAN environment. I can see how users might be accepting of that. In my company, however, for the past eight to ten years the vast majority of people, and now everyone, has had a PC on their desk or in their work station area. These people are very used to what they have. If they need WordPerfect, they get a copy. We've been liberal about that, but not so much any more. The pinch is on for software dollars.

We have a LAN up and running. Everybody uses E-mail. They like E-mail. We have been getting some applications on the network, but getting the people to use them and trust the network is another story. Any comments on how to get users to accept running things on the network versus locally where they don't have to worry about speed or the job getting done on time, and so forth?

MR. PETERS: Just in general terms, I think when you're talking about significant mission-critical systems, like Lanmark is in our environment, you need top level support within the company. Coming in at the middle of management levels in the company will not get the job done. You really need somebody at the top level saying, "I want to do this. I believe in networks. I believe in what the systems can do," and support it and make it happen. One of the things we talked about is that you had to rethink the way you do work. As soon as you start doing things in a distributed environment, you have to rethink the whole process. You can't think like you did 20 years ago. It really requires that high level support. If you don't have that, it won't work.

MR. PRINCE: Just a comment on user inertia. It certainly is no problem getting people to use the LAN because there it is. It's a real problem to get people past what I would call the initial stages of any software, and even something simple like word processing. They learn just enough to do what they want to do, and they don't seem to want to learn any more. We had one case that stands out in my mind where we converted our year-end appointed actuary reports, which are monstrous things, from one word processing system to another.

Someone, and I guess it was me, said casually, "Can we just reformat to get a blank line between paragraphs?" Well, there's a command for that. You do it once and then suddenly the whole document has an extra paragraph before each line. One energetic and well-intentioned person went through and hit an extra hard carriage return between every paragraph one at a time in a 30-page document. Then we had to take it out because it didn't mesh with the correct way to do things when we wanted to make other changes. I wondered, "Why didn't you look at the menus or look in the manual?" So inertia is a problem.

MR. KAYE: I've also found that another resistance some people have to using a LAN after they've used a PC, is they don't feel secure about their private information that they've had on their hard disk. It's a tough question. You can say, "So, take it off on a floppy, lock it up or take it home." In many situations it's just inertia, an excuse.

MR. JOHN N. CLAYTON: I'd just like to speak to a couple points that were brought up. Recently, over the last three or four years, we implemented a company-wide

## LOCAL AREA NETWORKS (LANs)

LAN at London Life, and we've gone through some difficult growing pains, I can say, with distributed processing. In essence, we ended up with distributed responsibility. The IS department, to some extent, disowned itself from a large part of the responsibilities that it took when it was doing the mainframe; so local support is critical.

The way we've handled it at London Life is we basically have broken things down into work groups that consist of two or three servers, or whatever, in functionalized areas, and we have a local support person there who handles the things like adding people to the LAN, if there are problems that someone has with a particular software package they can't get loaded up or whatever. That's the person you go to. For anything really critical or really technical, we also have a centralized support in the IS department that can help us out. With these people, we also have sort of the user group within the technical support people. They get together and that way they can share any common problems and provide some synergies.

The other thing I'd like to speak to is going to graphical user interfaces like Windows products, which is a point that Marc made. It can cause a lot of traffic on the LAN. One of the ways that we did it at London Life is we had, I think, a licensing agreement with Microsoft when moving to the Microsoft products, but basically everyone has their own copy of Windows and the Microsoft Office Products on their hard drive. That cuts down on a lot of traffic on the LAN, and it also gives you the potential solution, if the LAN is down for any period of time, to run the things off your hard drive. That might not be a feasible solution for a lot of companies, but it's one that seems to be working for us.

MR. LUKE N. GIRARD: Steve mentioned one of the values of putting your product administration on the PC LAN is it helps facilitate product development and lowers the cost of developing a system. Could any of the panel comment about the strategic implications of using the LAN versus a mainframe going forward? Have there been any studies done of market share or penetration? Does anybody have any intuition if there's no hard data?

MR. PRINCE: I've got some intuition. The problem we're dealing with in product development is really the inherent complexity of the product. I'm sure there are better ways to do it than we do it in the mainframe, but we're never going to have a slam-dunk instant access product development process because there are just so many details and how do you account for it, and what if he wants to do that, and what if he wants do this, that, and the other thing? I'm sure there's progress, but I don't think we're out of the woods yet in terms of product development taking longer than anyone wanted, but Jerry may have other thoughts on that.

MR. PETERS: I don't know if you're talking about the product designed from the actuarial standpoint or from the implementation standpoint.

MR. PRINCE: The implementation standpoint. Once you have the product design, the concept in place, and the ability to be able to deliver that product quickly.

MR. PETERS: Obviously, I think we have 20 to 30 years worth of experience in the mainframe environment to prove, in many situations, where putting dramatic new

## RECORD, VOLUME 19

products in the mainframe has not been a simple and inexpensive process. We have found that the networks and the PCs provide us tools and capabilities that just plain are not available in the mainframe environment. The language we use, the people say it's at least ten to one the productivity of a conventional language. We think it's probably in the range of twenty to one. Given a piece of functionality that has to be developed, we can do it a lot quicker.

Having the availability of tools like Lotus and WordPerfect in the network environment, which we use with Lanmark, gives us a lot of prepackaged stuff that we don't have to go out and develop. There's no doubt in our minds that getting products implemented is a lot faster in a network environment than we ever imagined in a mainframe.