

RECORD, Volume 22, No. 2*

Colorado Springs Meeting

June 26–28, 1996

Session 25PD

Technology and the Small-Company Actuary

Track: Computer Science/Smaller Insurance Company

Key words: Computer Systems; Futurism

Moderator: JAMES F. TOOLE

Panelists: CARY DAVID GOGGIN

WILLIAM M. WHITE

Recorder: JAMES F. TOOLE

Summary: Actuaries at smaller insurance companies face many challenges, including limited resources, broad responsibility, and limited opportunities to interact with other actuaries. This session presents case studies on the use of technology to solve some of the problems commonly encountered by actuaries in smaller companies. This session focuses on practical solutions and ways to maximize productivity.

Mr. James F. Toole: Bill White will share his experiences, which include moving from a mainframe to a fully PC-based environment. Cary Goggin is going to pick up where Bill leaves by off examining the use of database technology in a PC environment to address actuarial problems. Keep in mind that these examples may or may not be appropriate for your company.

Bill White is vice president and actuary of Interstate Assurance Company, a member of Irish Life North American Group. He currently heads a department of five people who provide all the actuarial functions, reinsurance administration, and policy form filing for the company. Prior to working at Interstate, he worked for a number of companies while enjoying different parts of the country. At his first job, he had the longest title in the company, which tells you something about the nature of the job.

Mr. William M. White: Interstate Assurance was originally a health insurer formed back in 1908. The company has a long history as being an industry leader. We

were one of the first companies to offer universal life (UL) back in 1980, one of the first companies to demutualize in 1985. We were the first of the Irish Life of North America companies, a growing group of companies in the U.S. Our parent company, Irish Life, is located in Dublin and has about \$11 billion in assets. We are one of the first companies to operate on a totally micro-based computer system.

Many people are starting to embark upon this process at the moment. It is something we started back in 1989–90. We have a number of visitors coming in to the company from throughout the country wondering what we do, how we do it, and how we can get so much done. The path that we took to our success really belongs to the entire company.

The company situation back in 1990 is probably not that different from a lot of other small companies. We were a life insurer offering UL, permanent annuity products. The company had about \$175 million of admitted assets, \$34 million in annual premium, \$29 million of surplus, and 101 employees. That was down from a high of about 125 employees. Like most small companies, we leased a computer system from a local vendor and we ran a major mainframe software package. For a small company this is a major expense, and this is one of the reasons we took a hard look at this.

We had a company president who had a new vision of the future, somewhat fueled by his success in the past. The vision involved the use of technology as a competitive advantage. We saw this as a way for Interstate to succeed in the future and this is the primary reason we took this path.

One of the major components of that vision was a new vision for an operating environment, something we called ECLIPSE. ECLIPSE stands for an enterprising competitive life insurance processing system environment. It is more than just an administrative system. For us it is an entire way of life.

Back in 1990, when this really started, we viewed our future as the following. We knew that we wanted to make the use of technology a sustainable competitive advantage. ECLIPSE was to be a major contributor to this process and also allow us to move into a low-cost provider strategy. This is important for a small company when you look at the resources of some of the large mutuals and ask, "How are we going to succeed?" The way we chose to succeed was to become a low-cost provider.

Throughout the process we had this next line in our mind. Our company president came up with this phrase, view simplicity as viewed from the right side of complexity. That might not make sense to all of you up front, but it really is a totally

different way of looking at things. To simplify everything outwardly was a major goal in this process.

What were the characteristics of the system we were going to build? Again, this was to be a sustainable competitive advantage for us in the future. We knew that we needed to incorporate relational database technology into what we were doing. We were going to offer a full array of financial service products in the future, so the system has to be flexible and adaptable. We wanted to integrate other related applications into this system, such as telemarketing, teleservicing, database marketing, electronic mail, producer and policy owner information systems, illustration systems, financial reporting systems, and valuations. This may not seem like much today, but back in 1989 and 1990, this was an extreme vision. We had a number of people telling us we would not be able to do it.

To accomplish this, we came up with ten commandments for success. We said back in 1990 that we will simplify, and we will make the appropriate use of technology. I want to stress that, the appropriate use of technology, not just any technology, but the right one. In order to make this work, it is important to communicate every step of the way so people know what is going on. Share ownership. Remember, I said this was a combined effort of over 100 people over the last five years. Everybody shared in this success.

We also said we were going to get it done. This was a key point. It is so easy to let these things go on and on. We set goals, we set dates. We also said what we would not do. We would not impede the company. So many times these efforts slowed everything to a dead stop. We would not impact the field. We would not imitate the old system and the old way of doing things. Everything was to be looked at differently. We would not transfer bad data. We were not going to repeat the mistakes of the past, and we would not perpetuate the use of paper.

What was the process that we went through? Again this was a company process. We had a steering group that held company-wide planning meetings. Each department was to identify specific issues relating to its needs, not only current needs, but future needs. What is it you would want if you could dream about the future? That is the key there, dreaming about the future. What do you need and want to accomplish four or five years down the road.? We had to go through a forecast exercise so that we would build the system of the future today. When you take the time and you go through this, and you have to go up to a steering group, and you have to give your vision of what the system is going to look like for your particular department or your particular needs, you need to think it through fairly clearly what it is you are going to do.

You must actively manage the entire process at all levels. I said communication is important. Internal communications is very important. Everybody in the company needs to be aware of what is going on. External communications are just as important, both to the parent company and to the field. Employee education is critical. We did not want to end up in the environment where only a select few people knew what was going on. We needed measurable check points and benchmarks for how to succeed at this, and unlike other companies, we chose a phase implementation approach to accomplish this. That decision meant that we were going to be running concurrent systems, some of the old, some of the new, at the same time. That meant this was a transfer process, not a conversion.

What were the issues back then? Some of them are still relevant today, but in 1990, you have to remember what the world was like. We needed to avoid technological dead ends such as the 1990 PC environment. I do not know how many of you were playing with them, but a 386 with an 80-meg hard drive was at the price/performance curve. Cost is a big thing too. You could buy power, but the cost benefit was not going to be there. We had a firm belief in the future of hardware advances. This is the thing that led us down this path. We always assumed that the hardware would advance faster than our needs would, but we also had contingency plans in case it did not.

We had a whole series of new roles that the company was going to have to undertake. Remember, we were leasing a computer system from somebody else, which meant they were the company operators. We now had to undertake that. We had to undertake hardware and software integration. Remember the massive array of PC back then, some compatible, some not totally compatible? This was going to be a major issue. We were also going to have to enter into networking, something that we had done in local area networks and a few departments, but this was to be a company-wide situation. Those are some of the data processing concerns.

In the other areas, we knew that in order for this to be a success the individual units were going to have to get very involved with the system. They were going to have to understand what it was we were doing. Much like systems analysts do in other companies, they were going to have to have ownership in the system.

Other issues that we dealt with were opportunities to simplify. This was a major goal for us. We wanted a solid-based system on which to build the future, which meant we were looking for places to start, we wanted things that we could change, modify, and adapt to our needs. We knew that we were going to have to customize processing logic for speed and efficiency. Remember I said back then that the 386 with an 80-meg hard drive was not exactly a real fire burner. We were going to

embed our company philosophy within the system. This may not sound like a lot to you, but it really had some major implications for us.

We said as a company offering UL products, we were going to be a portfolio-based company. We said that is our company philosophy. We were then able to go into the system and basically strip out anything that we did not think would fit with that philosophy. This helps speed things up immensely. In the same sense, everything else was looked at from how we were going to operate and everything was customized around those central ideas on how we were going to work. We had to optimize the month-end processing logic. We did a number of steps in that area to speed things up. We totally rewrote parts of the base system we started with, and we were probably one of the first companies to actually consider integrating our illustration system as part of the administration system.

We are unique, I think, in that we have only one set of values calculators. For people who sell UL there is only one target premium logic, and it is shared by everything. So you get the same number no matter where you go or what you ask, whether it is illustration systems, administration systems, or whatever.

For customization, we knew that we wanted to extend the basis that we were starting with. We were going to replace generic models with custom applications. We totally changed the written, in-place reporting requirements that the old system started with. We totally rewrote the commission and advancing logic so it was company-specific, again relative to our philosophy for doing things. The next one is probably unique within the industry. We have a product that had what I would call "shadow funds" or multiple funds, and we knew that we needed to integrate that with capability within the system. That was done. We are the only company that I know of that can actively have up to four additional shadow funds on an active policy processing every month. Again, all are consistent with our vision of how to do business in the products that we were selling.

What enhancements were we going to add? An inventory system. Remember Wal-Mart was getting big back then. Their success was built on point-in-time inventory, being able to click on something and get the products sent out. A decision-support system, which would give everybody throughout the company the ability to do on-line inquiries and access to important data. Again, bearing on that relational database technology we were hoping to embed a producer profile system, which would give our marketing department added levels of detail, any producer at various levels, and a new business front-end. There are a lot of repetitive tasks that go on when you are keying business into a system. You enter something here, you enter something there, you enter it all over the place, and you do not need to. So you can look for places to optimize your business.

Other general concerns: all employees need to be involved. Remember our shared ownership concept. Communication at all levels. As I said before, this is a transfer process, not a conversion. We never used that word, we never took that attitude towards it, and we were only going to transfer good data.

Where are we? Where did ECLIPSE end up? A quick jump ahead might put some of this into perspective for you. We ended up with a flexible design for our system. Responsibility resides with the individual and the individual departments, not with the data processing department. That means you have to do certain things. You have to empower people to be able to use their systems to understand it. This is the current ECLIPSE structure. This is how it exists today. It is a modular-based system.

Surrounding everything at the moment is something we call One Link. One Link is a very comprehensive shell which allows both internal and external access to all of the functions that are listed underneath it. With One Link, you can be anywhere and effectively be in the office doing your work.

Underneath One Link is a Windows graphical environment at the moment. We found this to be the best way to get people to be able to use the information in the system. The Windows lays on top of everything else. Underneath that you see the main modules of the system. I say modules because we have expanded. We replaced certain things as we went along. The first one is just the generic service modules, the other one would be the database marketing module. When I talk about wanting to add the DSS to our system, that's the decision support system. We currently run two administrative systems underneath it. We run both the leverage and the PDM base system. Underneath that are the tools that all of the employees have access to get at their information: Word, Excel, and Access. Underneath that is an e-mail system tying everybody together, and underneath that would be the rest of the communications: network, data, voice, fax, video transfer, and everything else.

That is the company-level deal, but what about the actuarial department? What were we dealing with throughout this period of time? We said that we needed to have the ability to add new products quickly. We needed a timely valuation. We needed to change our reinsurance administration. We needed to build product management capabilities. We also needed to build the financial forecasting capabilities. In 1980 and 1990 forecasting tools were just coming around, the valuation actuary concept was coming about, cash-flow testing was going to come about. We saw this as a need we must build on.

We knew we would be building subsystems, and that raises a whole level of other concerns for actuaries. Documentation, readability, and support is critical. The

work that you do here is a company investment, and it should not be a one-time investment, it should be something that carries forward into the future. You have to be adaptable to the changing times. In a small company environment, you cannot afford to go back and start over from scratch time and time again just because you have had a personnel change.

Actuaries are not good programmers. We can code solutions, but that is not programming. Programming is called maintenance. Eighty percent of your work in programming is maintenance of the code, extending it. We knew that we had to be good at this. We wanted to build on a solid foundation. You cannot keep switching all the time, especially in a small company. You need that continuity. There is a trade-off in speed versus ease of use. Again we started this back in 1990, so our tool selection was a little more limited than it is today.

What did we decide on? We decided on BASIC and X-Base as our standard from which to work. That generally meant DBase files. As far as actuarial programs, we were going to use BASIC as our ongoing language. Other choices were APL, C, or whatever, but BASIC tended to be the easiest to learn, the easiest to maintain, and at that time was probably one of the most adequate tools to go with. That is not to say we do not have some special applications that are written in some other languages. We still do. We are still dealing with such issues as: what are we going to do? Are we going to rewrite those? Are we going to leave them? and What is going to happen? Behind all this, remember the ECLIPSE diagram. We have the office tools that you think of today, spreadsheets, those are also in there. Spreadsheets, database, and a basic programming are standard.

What is it that we needed to do? We knew we needed to put up new products quickly. We had to build rate-book interfaces so that we could transfer data very quickly. Remember, we were all going to be on the PC environment. This meant that you could code a plan inside the system in a matter of hours, not days or weeks. We needed to have the same transfer of information to our illustration systems. Again, our goal was continuity of the data and speed. If you do not have to worry about somebody keying in 200 rates and then having somebody else check them, it makes a big difference. Policy pages were integrated with everything. For us, for UL projections, you have to have the values. You want to make sure that the illustrations match the values and the policy pages, and match all the calculations inside the system itself.

Let's discuss financial reporting. This may seem aggressive to some of you, but our president believed in monthly statutory reporting. We have a close eight days after the end of each month. That by its own task is monumental, but when we take it a few steps further, we do line-of-business statutory financial. Within a line of

business, we will also report by financial plan. We will also do something which I am not sure many of you do. We will do a first-year versus renewal statutory financial. Working all the mechanics for this took time and effort, but if you do it and you build on what you have, and you embed it into your systems, you can leverage off this in ways you will not believe. Materiality for us in a monthly financial could be \$5,000. We want to check it down to a plan level. We want to know where it is, where it is going, and what is happening.

Quarterly embedded values is a European accounting standard. This is how they report their earnings. This is similar to a cash-flow projection without assets. You have to do these quarterly. Cash-flow testing was emerging. We have to do that at least annually, and this year we are fortunate enough to get into generally accepted accounting principles (GAAP).

What did this mean for us? It meant that we had to take a hard look at our reserving systems. We had to have something that was fast and efficient. With eight days to close, you cannot spend a lot of time tracking things down. Again, our product line is UL, term, and annuities. For those who are in health, some of this may still apply. I am going to run through some of these things, but the point is you want to take a hard look at what you are getting, what you need, and what is the best and fastest way to do it. We decided that we were going to rewrite our annuity reserving system. We were going to incorporate the commissioner's annuity reserve valuation method (CARVM) calculations. We would do this through monthly extract files that would be available to the actuarial department. We are in essence doing some of our what you would consider an information systems (IS) function in running reserves.

What did this give us? It gave us the ability to implement the option base reserve for annuities that were implemented last year in about six to eight weeks, where some companies were still guessing at it four and five months later. It also gives us the ability to project future reserve increases. In this extract file, we can say, what if we credit the current rates for another year? What will the reserves be a year from now? You can use this when you are doing your financial plans for the next year.

For term products, we decided we would use what came with our base system. This seemed to be the most efficient and most flexible. We were then left with UL. In 1991, we looked at our available options, and that is the place you always start. What options are available to you? You use the base system calculations, which we would have to fully test out during the transfer process. Information systems resources were scarce, so we ruled that one out. We then looked at the vendor-based software in 1990-91. There were not a lot of choices, there were primarily two that were out there. The third alternative was in-house development, and

again, consistent with our company philosophy, we chose that. It allowed us to be in control of the process, and at a lower cost. Again, the data transfer aspect was critical for us.

We went through a monthly extract process. We were able to implement the UL model regulation at a time when other companies were just starting to think about this. We built the system for expansion. It takes about fifteen minutes to add a new plan into our system. Why? Because we control it. What drawbacks are there? We have to run it every month, but when there is a problem, we can find it.

Remember I said reinsurance was another actuarial concern of ours. We had been using a third-party administrator based too far away to go to conveniently, which meant we were always having trouble reconciling. We were having major expenses leasing the system from somebody else. We said, we cannot do this. We are a low-cost provider, what are we going to do? We need to bring this in-house. What were the options? Find a system externally. We were fortunate enough to be involved with a reinsurer who offers a reinsurance administration system as long as they are your reinsurer. We opted for that. It is an X-Base system, and remember I said before that we opted for standards that were in X-Base or DBase components. This gave us an added level of access to detail. It also meant we were comfortable with the file structures that were out there, so we were able to convert in four months what other companies were taking six, eight, nine, or ten months to get through the process.

We did acquire a couple of other things. We basically needed to build a new business interface. Again, we wanted good data. The ECLIPSE system automatically sends information to the reinsurance system. I do not have to worry about files being flagged, people being missed, or whatever. We needed a general ledger interface. We needed information going back into the system. We also needed audit functions, which are probably the weakest aspects of most reinsurance systems today. Again in our world, we can get at information in the PC environment. We can build the audit functions that are missing everywhere else.

Actuarial concerns and information are critical throughout the whole process, and on a PC-based environment, there is good reason to spend time worrying about this. If you have good information you can do everything else.

What were our sources that we were going to be dealing with? We talked a little bit about the reserve extract files we were going to get, because we were generating the reserves. We can get information out of the administrative system. Remember, in the One Link there are tools for pulling information out of the system. We had the reinsurance administration system files to deal with. Again, a common factor

running through there is they were all X-Base systems. We can get information from all those and link them together.

Where we were going to put them? One composite source to hold all the information, something we will call an actuarial database. How was it going to work? Quarterly, we would update the database with all active policies. The information was always accurate for every policy still active. What would we save? We would save the terminated policies forever. What do you get when you do that? Small data storage on the terminated records that never go away, quarterly update of the active ones. You get a very good data source that you can use for many things. Consistency of the data is very important. If it is bad data, you cannot use it. If you are trying to reconcile bad data, you are wasting your time.

All actuarial departments, I don't care what line you are doing, do experience studies. For me that means lapse studies and mortality studies. Again, I have a data source. I have an actuarial database. We are able to do the lapse studies quarterly. Right now it takes probably less than two hours to run the lapse study that gives me lapse rates for the last four or five years. What can I get out of that? Unbelievable information. I can go down to producer, by product, or by issue year if I want to. Why? Because the data is out there to be able to get at it.

Our most recent effort in this area is an on-line inquiry system for the rest of the company to get at. They can point and click under Windows to get at that same information. I will still produce management reports, high-level reports, but anybody in the company can find out what is this particular agency doing in UL issued in 1992, and it pops up. When armed with that kind of information, people can and will make the right decisions.

The mortality study is another experience study we do annually. It's source is the actuarial database and our claim system files. We are fortunate here that the claim system is a also DBase-oriented file, and it is very easy to merge the data from the two databases.

I alluded to product management a bit. We produce in-force and new business studies. These are management base reports that we distribute to all key managers. What do these things have in them? Essentially all of the relevant detail about the products we are selling that you would want to know, as well as the path information. You can trend what is going on. For example, we have one product where the change in the male/female distribution has been slowly changing by 2– 4% per year every year. This sort of information leaves you to wonder what is going on, and you can then start to find out what is changing, and why this product is being sold this way. Another product we have for UL has been progressively changing

from a level death benefit to an increasing death benefit. Why? What is doing this? This type of information is invaluable when you are trying to manage your line.

The ECLIPSE scorecard. How well did we do in what we set out to do in 1990? We are still adding features today, but the basic foundation is right for us. We have integrated components in place today. We do have a database marketing operation in place. We do have electronic mail. We do have producer and policy owner information systems in place. Our illustration system is integrated with everything. Our valuation and financial reporting systems are solid and in place. We still have to expand on them as the regulations continue to change, but the information and the structure are there to allow us to do that.

What has all this done for the company? In the end, that is the relevant impact, not whether you can build something and make it work, but did it have a positive impact on what you were doing. In 1990, Interstate had about \$3 billion of in-force life insurance, it is now over \$6 billion. Our number of employees was 101 back then; it went to a low of about 82, and is now up to 95. Policies in-force per employee is one way of measuring productivity. It is not necessarily the best way, but it is one way. In 1990, we had roughly 340 policies per employee, in 1995 we had 881, and in 1996 it is over 900 policies per employee. This number continues to go up. The other relevant financial measure I would say would be assets for the company, which went from \$175 million to over \$1 billion.

In closing, I want to stress that this is the overview of what Interstate did, based on our idea of the future and our vision of what we wanted to become. Some of it may or not be relevant to you, but the point is if you take a hard look at the process we went through and sit down and ask, how does this apply to me? What can I take from this? What is it we can do? Then an entire company can move forward, and with that the individual departments as well.

From the Floor: Bill, how do you coordinate the changes in modules that are going to occur?

Mr. White: We do not change the modules. When a module is changed, that is actually done down in the data processing area. What we do is we have the information to work with. In terms of reserving, data processing does not care what your reserving system is or what it does, they just care that it runs in a normal company environment and they care that you are not coming back to them and saying, rerun it. So we do what we want to with the reserving systems. They do not care. When you take on the responsibility and realize that you are never going to be able to give it back to them, you are stuck with it. You are a data processing department from that point going forward. You make that conscious decision based

on your ability to get good data, and your confidence and your ability to support the programs that you are going to write to calculate the reserve. If you do not have that basic tenant, then you do not want to go down that path.

From the Floor: How did you overcome the logistics from the traditional years?

Mr. White: You have to realize that the vision came from the top down. Our company president was in this. He had been at another company where they had successfully implemented a similar sort of strategy on a mini system.

From the Floor: What happens if you do not necessarily have support from the top?

Mr. White: I think it will be easier in the future as more examples of success come to the surface. One hard way to look at that is to simply look at your cost structures and say can you continue to do what you are doing now and survive. Will what you are doing today carry you forward another five years? If the answer is you don't think so, then you have to look for a new vision.

From the Floor: What has happened to your data processing staff?

Mr. White: I think the size of the department has remained roughly the same. It is not a large staff, it is probably fewer than 20 people, but they have taken on so many other tasks that are not traditional data processing. Again, we are now in a multi-company environment with offices in a number of places, so we are doing wide area networks and video transfers. At the same time we are continuing to push the One Link and ECLIPSE systems forward with the changing technologies. I do not think you will see any staff reduction in your data processing department.

From the Floor: Can you talk about what you did in approach to simplify?

Mr. White: The basic thought was everything we do should be challenged. Why are you doing what you are doing, especially if it is complex? Is there a simpler way to do it? Our president had a favorite line, which is, if you are going to do something unique in the product, do it in the first year, because you do not have to live with it after that. Once you pass the first year it is done. If you have some sort of special refund feature on your term or whatever, do it in the first year. Again, you can take everything that you are currently doing and say, does this make sense? Can I live with it? You are right, it does become a Zen sort of thing, just as becoming a low-cost provider becomes a major driving force. Common ownership of the process and the end result is important to make this succeed. If you do not have it, you will not.

From the Floor: Do you keep historical data on your active policies?

Mr. White: No. We do keep information on the active policies. It is in the database. Cary is going to talk more about database structures and what you put in things and how you utilize it. Without me going into details of which fields we keep, that is really determined by what you are doing with it. If you need to know who the current producer is versus who the writing agent was, then you have to design your information around that. If you do not care, you do not.

Mr. Toole: Cary Goggin is the directing actuary for American-Amicable Life Insurance Company of Texas. That is a subsidiary of PennCorp Financial. His responsibilities include all actuarial functions, including product development, financial reporting, compliance, illustrations, cash-flow testing, and financial projections. You know if he is doing all that, he has to have some support on the data processing side to get all that done. His work history includes five years at Interstate-Assurance with Bill. He took what Bill has done and has gone in a slightly different direction. He is currently working on his MBA at Baylor University.

Mr. Cary David Goggin: How many of you use a PC regularly at work? Almost everybody. Did we all start using PCs at the exact same day and to the same extent? The wave occurred over a couple of years. We gradually got used to it. We used them a little more and our capabilities increased, and it took a little time. There is an awareness and a learning curve that we go through when new technology or new combinations of technologies become available. Sometimes when technologies first appear there is really no way to know the profound impact that they are going to have in the future or what the future holds.

I am going to cover three things. First a little on the productivity increases which are realistically achievable today with some of the PC database software which is available. Second, I am going to try to give you an overview of some of the things that I have done. Third, I am going to try to give you a feel for what types of things you need to get started down this path if that is what you choose to do.

Regarding productivity increases, back in the 1980s, many of us actually experienced taking a productivity leap in areas such as financial reporting. What software tool did we use to do that? We used the spreadsheet. In financial reporting, for example, we identified all our inputs, from paper valuation runs and various sources, and we built the spreadsheet models. That enabled us to do our financial reporting in a fraction of the time that it used to take. To an extent, that is still done today. Some of us get some summarized information from our administration systems downloaded to us, but the financial reporting paradigm is still roughly the same as it was ten years ago.

We can take another quantum leap in productivity by moving from a summarized valuation run/spreadsheet paradigm to a nonsummarized/database paradigm. Financial reporting represents only a small fraction of the tasks where we can achieve this type of quantum leap in productivity. For example, what do we do when we need reserves, face amounts, policy count, or some other breakdown of our in-force when our computer runs do not have that split? We have to spend time flipping through runs and piecing it together, or worse yet, you have to make an IS request, which can take hours or sometimes a day or more.

With an in-force database that has information at the policy- or administration-system-record level directly available to you, what previously took hours or days can now literally take minutes or seconds. That is tremendously significant. When you imagine every record sitting in a database on your hard drive or on your local area network (LAN) at your fingertips and combining that with today's database software and using our abilities as actuaries to apply this to some of the challenges that we face, the potential is truly exciting. What we need to do is expose ourselves to a few of the possible applications of the PC database in our own individual situations. Once that happens, the floodgates will begin to open and we will begin to realize all the possibilities of what this type of software can do. The best part is that this can be done relatively inexpensively and you do not have to be an expert programmer to pull it off.

I would like to describe a sample of the flow of the data from the administration system or whatever source of information that you have to the database software. The original data will more than likely come from a policy master file or valuation extract. That can vary depending on your individual circumstances. The data flows from the mainframe environment to your hard drive or LAN via a download in the form of ASCII text files. I have several files downloaded each month. They are basically separated between base policies, riders, and supplemental benefits. The base policy, rider, supplemental benefit split is logical because of the way the database software associates or links data between different tables. It is very easy to pull up a base policy and associate it with all the riders and supplemental benefits that exist for that policy.

After you have the data in the text files, you use a file-import-type function to get the records into the actual database. When I get the data to this stage I do two things. One, I compress the data with the compression utility for back-up and for storage, and I also copy the tables to our LAN for other users and my own future use. Once I have the data in this format, I can already begin to perform tasks with the data or ask questions of the data in a free-lance fashion. I will talk more about that shortly.

For tests that require repetitive or more specialized processing the current month's tables are added to the tables used by a database application, which is nothing more than a program. Once you have these tables available to this application, you can do many things. I am going to explain some features of an application just to give you an idea of some of the types of things that you can do.

An example of a main menu is what an Actuarial Database Application Menu (ADAM). I will give you a description of how you can use this to build this type of a screen. To run this, you click on any choice with your mouse. The first button is labeled "in force". This button, for example, will perform all the steps that get your data from the ASCII extract to the database application and everything else in between, including the compression and storage. Another button brings up a policy status including riders and supplementary benefits. An actuarial department status can be tailor-made to whatever type of things you need. You can have quite a bit of information on it. It's all up to you what you want to put on it.

Another choice off the main menu is a reserve activity form. This screen, after you have made some check marks, deciding which sub block of the business you might be looking at, will actually create an extract required by a PC valuation software program. The database software, by use of some code, picks out all the fields the valuation software program needs, puts it in the correct format, and creates a text file. You can then turn on the valuation program with this button and run the valuation. The valuation software then creates its own output files. I also have a button that imports the results back into the database for further use.

In my particular situation, only some of the policies are calculated on the PC. There are still a lot of reserves that are calculated back on the mainframe. So to get the final reserves, the button also takes the calculated reserves from the PC software, puts them into a final reserve database, then searches for policies that did not have any reserves calculated, and simply copies in the mainframe reserves into a final file. Then this file can once again be used for various things.

Another button on the reserve activity form will take the final reserves for the month and then add them to a history file that includes fields like death benefits, statutory reserves, GAAP benefit reserves, and a few other things. Again, I am explaining different submenus of the main menu. We have a screen that is a historic information area, and for example, if you wanted to look at an individual policy, you would get reserves and key valuation information by month at the policy level, and once you have that, you can easily create trends at any level or subsection of the in-force that you want. That is very powerful.

I also have some buttons that perform various types of tasks. There is the company illustration system. It is just a convenience to have it there. Keep in mind that this all can be done in a LAN environment and this main menu can be made available to other departments in an Intranet fashion with different people having different levels of access. I can also take tables of rates or factors that are developed by pricing software and convert them into the format required by our administration system, so it can be uploaded directly. You do not have to type in rates when new products are developed. That can save a lot of time.

Other miscellaneous sub menus include programs such as commutation function generators, and direct access to the department's reports and procedures. Anything can be put anywhere. As you can see, many of the things you can build can be specialized to your own particular needs.

Now, how do you get started with all this? What do you need and how does this all work? Regarding the hardware, to run these types of software, which is Windows based, and for all this to run smoothly in the real world, which can involve turning software on inside of software, you are most likely going to need a PC with about 32 megs of RAM, a hard drive or LAN with enough free space to hold your data several times over, and I would say at least a 100-MHZ processor.

Here's a brief description of how the software is organized. First, like any database software, you have tables. You have the fields going across, the policy number, the issue age, etc., and the records going down. An important part of the software is the query. Generally you ask a question of the software by showing it an example of what you want. Let's say you wanted reserve totals by state. You would do something like the following. Click the menu choices or icons (just like a spreadsheet) FILE/NEW/QUERY. Then you pick the table that you are curious about, and then you see a screen with all the fields of the table represented. So to get reserves by state, for example, you would type CALC SUM by the word RESERVE and put a check mark by STATE. You then press a run button [Enter]. You will then see a table that shows the RESERVE totals by STATE. It is that simple, and the answers actually come in the form of a table. You can print it, you can save it, you can send it to a spreadsheet, or whatever you want to do with it.

If we were to put a check mark by STATE and PLAN, you would get reserves by state and plan. Once you finish with the query, you have the option to throw it away or save it just like you saw. You can also use more than one table in a query to do more complex functions. For example, let's say you wanted to do reserves by region where region is a block of states. You would have another table that has all the states and what region they belong to. If you want a state by region, you would have to link the two tables by putting a mark (say an X) in both STATE fields to show

they are associated. Now enter a check mark by REGION and use your CALC SUM command in the RESERVE field and you have it—reserves by region.

Another important part of the software is the concept of the form, and the form is where we begin to diverge from these free-lance activities, such as the queries we just looked at, and move toward some of the structured activity, including actual programming and application building. What you do is start with a blank screen. There are tool bars that are available. You pick the "button tool", for example, and you drop it in the form. Then you put pieces of code behind the button so that when you press it, that piece of code executes. It does not have to be a little piece of code; it can be a sophisticated program if you want. The buzz term for this kind of stuff is object oriented programming. You are putting pieces of code behind different objects instead of writing a program in a linear fashion, like many of you are used to doing.

I find it fun to do because the development of a form is almost like painting a picture. You enter the code after you paint the picture. The types of things you can do when you press a button are unlimited because you have a whole programming language behind it. You can do things like have the software execute ten queries in a row that you previously wrote by hand, for example. You can think of that like a macro on the spreadsheet.

You can also create program code almost like BASIC with loops. Those things can be required when you have to do complex functions that you need when manipulating data. When I started throwing down pieces of code I used the help area of the software, and it has examples of code in it. If I needed to figure out how a DO LOOP worked, I just pressed Help, searched on the word DO, found an example, and then I just cut the example out and pasted it into my form, and then modified it so it used my tables and fit my situation. One advantage to these higher-level programming languages is that you really do not have to be an expert programmer to do this kind of thing.

I will talk briefly about the data. One of the biggest components of implementing a database like this one is getting the data you are going to need, and this can take some time. The first step is deciding what data you would like to have. It is going to have to include all the fields for every sub-activity that you are going to be working on now and in the future. A list of fields has to be created. Then for most of the fields you are going to have to have a way of testing the data and verifying it: comparing totals to valuation runs, checking the data for oddities, and looking for blanks, for example. You are also going to need to figure out where the fields you need are in your administration system. You are also going to identify a list of fields

that make your records guaranteed unique. Those fields are grouped together in the software as an index.

That index has several uses with this type of software. It is used to relate one table to another and improves the speed of many functions. Splitting up your list of fields into tables is another consideration. I explained earlier how we had a table with base policies, riders, and supplemental benefits. In general, the fewer fields you have in a given table, the faster the processing will be. In my case, I had too many fields once I got the list done with the base policies. I had to break it up into two. I took all the base policy fields that were common and used frequently, put them all into one table, and then took all these extra fields that were not used quite so often and put them in another table. Why? When you develop these little programs later and things have to execute, a lot of your programs are going to execute faster.

To wrap it up, why is all this important? For most of us, there is a trend where there continues to be higher demands on our time and there is not always an equal compensation in the resources that we have available. Also a potential advantage exists here over the larger companies. They would have a harder time implementing and using these types of things that we talked about. Once you get over about one million records, you have to become very efficient with your programming for this all to work. For companies that have two or three million records, I think they would have a hard time doing this and end up being more productive doing it their old way. As software and hardware continue to advance, the number of records these database applications will work with will increase.

The caveat that all of us have been talking about is to keep things somewhat in perspective. Before we plunge in and make use of this type of database we all have to ask ourselves if this is the correct thing that we should be doing for us and does it mesh with our environment. Each of us is going to have to answer that after we analyze our own situations. If this is the type of productivity increase and technological change that is good for you, then I think you should look into it, because it really can be done.

From the Floor: This is all really exciting because we are not a small company. We have close to 2,000 employees and my struggle has been just been getting data. It appears that I'm heading down the same trail that you have headed down, and what I'm struggling to do now. We're not on Windows 95. Have you had any success in linking database and spreadsheets together?

Mr. Goggin: Yes. This happens in the financial reporting area, because when you have to send information to the accounting area when the financial reporting is done, well, they are not doing this type of thing yet. Even though a lot of the

processing can be done with the database software, at the very end, you still have to turn it into a spreadsheet, or if they still accept paper, you can print it off and give them the paper. Perhaps they are on a LAN and they can accept the spreadsheet. You can give them a copy through copying a file or sending an e-mail. The software does work very well with spreadsheets back and forth, and doing some processing in the database, sending it to the spreadsheet. If you do not think you know how to do it in the database, go ahead and finish the processing or manipulating in the spreadsheet area. Certain softwares may work better with certain spreadsheets. Some are made to work with each other better.

From the Floor: This may be too specific, but we use Excel and Access.

Mr. Toole: Excel has its own database facilities and functions, and Access takes it even higher. They are really streamlined. You can go from here to here to here with no problem, and then even the Word program has some simple spreadsheet functions, so that you could go from Access to Excel to Word very simply. Depending on how much you are working, Access can handle a lot of records.

From the Floor: Yes, it can. That's how I got into Access. Again this may be too specific, but is there a reference that you could give to me to try automating this linking of things. That's what I can't do. I can do it manually. I can export it manually, but I can't get something automated to run.

Mr. Goggin: I do not use Access and Excel. I happen to use their counterparts—Paradox and Quattro Pro.

Mr. White: My suggestion to you would be go browse the bookstore, and you will eventually find a book that will address it in the language that you understand and the examples that you think relate to you. It may take hours, but somebody has written something showing you how to do something similar to what you want.

The other thing is do not limit yourself to insurance personnel as far as finding information. These tools are widely available everywhere. You can find experts in any city who will be willing to spend an hour of time for you. They will charge you for it, but they will show you how to do what you want to do.

Mr. Goggin: I know there are commands in Paradox that will print it directly into a spreadsheet. That is just a built-in feature.

From the Floor: I think there is one in Access, too. We tried that a couple of weeks ago and we did not have a problem.

Mr. Goggin: With my software I did not have a problem with that.

From the Floor: I have a program in that language, but maybe that would work. Our companies are mainframed right now, and I went out and I hired myself a 19-year-old, first-time college student.

Mr. Toole: It is a link situation for everybody. These college kids are smart, they are motivated, and they have some great ideas.

From the Floor: What about the actual consequence of this system when you calculate the reserves on the fly on a PC that you had on the mainframe? When you have to try to explain to an auditor that you are using this software program, you have a hard time showing an auditor a program.

Mr. Goggin: With the PC software that we are using right now, the only reserves we are doing are UL, and the software that we use in my mind have incredible audit functions. It can take you right to the final reserves, step by step, all the way back to the interest rate, and it has information available at the policy level, and then you can choose whatever months you want.

Mr. White: Again, when you look at this, you have to consider all the options as far as how you are going to do it. If you were going to build your own system, and I am saying that you probably could, that is one of the key things you want to look at, how am I going to build an audit function? How am I going to convince the auditor and the examiner, but also myself, that I am getting the right numbers coming out? You will want to build products, not only audit functions on the reserve count itself, but financial audits. I would suggest that it is something most anybody, with the schools that are out there now, could probably take on and do, but you need to approach it from the right mentality. You better build your own audits in there, and in the end, I think you will probably be happier.