RECORD, Volume 27, No. 3*

New Orleans Annual Meeting
October 21- 24, 2001

Session 142PD
Actuaries In Computer Science

Track: Actuary of the Future/Computer Science

Moderator: PHILIP J.T. CERNANEC

Panelists: JAMES W. LAMSON
BRIAN S. REID
ROGER W. SMITH

Summary: There are a number of actuaries in nontraditional roles in the computer science field. Some of these actuaries make their living programming, managing systems, or managing data.

The panelists discuss how they got into computer science and explain the various types of activities involved in their jobs as actuaries in computer science.

MR. PHILLIP J.T. CERNANEC: As I mentioned, this is Session 142 and we have three esteemed panelists speaking about actuaries in computer science. Our first panelist is Jim Lamson. He's from Actuarial Resources Corporation in Kansas City. He has a BA in mathematics. In addition to that we have Brian Reid from Milliman USA, involved in the sales and marketing of the ALFA system and his territory includes some international locations. He comes from Connecticut and Brian is an accomplished product development actuary. Roger Smith is from PolySystems Inc. and he's led the development of that company and some of the areas of reinsurance, evaluation administration, and has mostly been involved in what he says is saving money by spending money with PolySystems. He's been with PolySystems since 1981. As you know, many of their programs are involved in the development and sale of software to make the actuary's life easier.

Part of my background at the moment in time includes Cap Gemini Ernst & Young. Cap Gemini Ernst & Young is involved in management consulting and information technology services, all the way from custom built to outsourcing or application management. I'm also involved in enterprise work. My particular areas are in the insurance market around sales, service, marketing—anywhere from strategy to enterprise application implementation. First off, maybe you could talk a little bit about your background. In your childhood, what made you "go wrong?"

MR. JAMES W. LAMSON: Phil, I guess I'd have to say that I went "bad" in college as a lot of people go bad in college. That was probably my first exposure to computers, because back in those days, and I don't mean to make it sound like such a long time ago, but my first exposure was in the early '70s. You could get as much done easily in an afternoon today as it took a month to do back in those days. I would have had a double major, but back in those days all the computer science courses were in the mathematics department, so I had a boatload of mathematics credits by the time I graduated. I had around 70 credits in mathematics, but half of those were in computer science. Probably all of you in this room share this view that there's just a magic associated with computing and with the programs, and making a computer do the things that you want it to do. I got caught up in all that early on.

We had a lot of fun in the mathputer room. We devised a scheme one day where some of us were going to go to Durbin, the local horse-racing track. Of course, that was before cell phones and so forth. We would phone some information back to the computer room at college so we would have a program. We were going to try to make money doing this. Of course, I didn't have any thought that it was probably highly illegal, and fortunately we didn't actually pursue it, because otherwise I might be looking at you from behind bars. Anyway, that's how I got started. I've had a fascination with computers all my life, which is just a lot of fun.

MR. BRIAN S. REID: I didn't go "bad" until much later in my life, specifically six years ago. In my educational background, I did everything I possibly could to avoid computers, whether it was high school when they first came out and I was exposed to them, and college, with the punch cards and all that, I did everything I could to stay away from that. Then the whole idea of sales still makes me laugh to this day, because I just remember being in little league and being so averse to ever thinking about selling anything, that I paid my younger brother to go out and sell all the cookies for the fund raiser. It's certainly been a big change and I'll talk about that later.

MR. CERNANEC: I'd like to follow that comment up before we lose that train of thought. I think actuaries in particular have a certain resistance to being sold, and I think it's a perception thing that you can see it coming a long time before it delivers. For most of us actuaries, we're kind of considered a hard sell, but what you really need to recognize, and I'm sure you all do, is that you're selling things every day. When you're making a presentation to the board of directors or just to your boss, or you've done a lot of work and you're presenting it to someone, you're selling them on the idea that what you've done is valuable and good and correct. For example, selling actuarial software is very similar to that. We're not out there selling new cars that have defective transmissions and we're going to try to put one over on somebody. The idea, I think, of selling anything is a matter of believing in your products and it's easy. It's not a matter of selling used cars.

MR. ROGER W. SMITH: I have a couple of comments in terms of when I first went bad. I think I had worked as an actuary for probably three or four years doing a lot of computer projects before I first realized that not everybody did that. In the environment that I started out in, everybody did some of that, so I was shocked, stunned and surprised when I first started coming to meetings and talking to other actuaries of my generation and found out that they didn't do that.

MR. CERNANEC: From that perspective, Roger, was there any triggering event that you could go back to where you could say you moved into more of a computer science position as opposed to actuarial practices?

MR. SMITH: I wouldn't necessarily say it's one or the other. I like to think I can do a little of both. I would say that the thing that got me looking at something a little bit differently was after I had gone through the exams. This happened to me and I'm sure this happens to some people. You might look at your boss's position and think that you would like that job, not tomorrow, but I would like to do that job someday. I can remember thinking that my boss's job didn't seem to be all that much fun, and that he didn't get to do very much. In that company, that was a good job. It was kind of a designated frog eye. Any time anything really bad happened, he tended to take the fall, so it wasn't something I really desired to step into.

MR. CERNANEC: Since you still have the microphone in your hand, Roger, maybe you can expand a little bit more about your education, your background?

MR. SMITH: I did take quite a few courses in computer science, and at that time my experience contrasted a little bit from what it is today. I have a daughter who is at the University of Illinois and last semester she took her first computer programming course. Now she has Visual Basic on her laptop, full diagnostic. As soon as she types in some sort of a syntax error, it tells her about it. Now, my first involvement in programming computers meant going to the computer center in the middle of the night. I mean it was best to go about 2:00 in the morning, because if you were a beginning student you had to go out there, punching cards in the middle of the night, watching through some screen to see your job go through the execution and just hope that the printer was not broken as it was about 50 percent of the time. That's what life was like getting started then. It hasn't been that many years since, but there has been quite a dramatic difference in learning computing or programming aspects of the computer today.

MR. GEORGE L. ENGEL: I have a question for Jim. I see it says you have a bachelor of arts in mathematics. I'm just curious about your degree. Apparently you did that with computer science and mathematics while you were in college back in the '70s.

MR. SMITH: That's just problems I went through and the way the degree was. I took a lot of mathematics and computer science classes. Actually the beginning course was a made-up language called MOHAC, a machine-oriented something or other and I don't remember the name. I certainly have German and some other courses on the art side of it and the requirements, but I wasn't heavy duty into physics or that sort of thing.

MR. CERNANEC: Thank you, Jim. Brian, any elaborations on your orientation, your background?

MR. REID: I was fortunate enough to find out about actuarial science as I was entering college, trying to be an engineer and was able to shift just as I got there, so I did get to major in actuarial science and start passing exams in school, which was a good head start, and then got into my first rotations. I got into the development and use of actuarial projections, modeling. As time passed I got to start some vendor systems. So going from the home-grown to the vendor package, being exposed to the advantages and disadvantages and so forth, that just kind of fed into where I ended up.

MR. LAMSON: My triggering event actually consists of two pieces. I had an opportunity to go on some of the sales calls with the brokers, and I recognized pretty early on that they were making all the money and I was answering all the hard questions. I decided that I would really like to use my education and my background to get into a sales role. So I actually started to investigate moving to a brokerage outfit as the token actuary. While I was in the process of doing that, my wife, an accountant, was laid off from Aetna at the time and became a financial planner. She succeeded and I went with her to the social events for their agents, and in talking to a lot of the agents, I noticed they're also making an awful lot of money and they really don't know much at all about most of these products.

I again was trying real hard to figure out how I could apply my background and education in the sales world. I actually went and I was trying to get to sales more than computer science and that's when an opportunity came up to move to SS&C PTS. They had just bought the Chalke operations and sold PTS, which was the vintage system I happened to be using at that time, so I was able to dovetail into exactly what I wanted to do, which was to move to technology and computer science.

MR. CERNANEC: What put you into it?

MR. LAMSON: For sure, I was considering a couple other possibilities. There were some guys in town from the CPA firm that were telling me I was just perfect for their operation and why don't I come along there. There were a couple of reinsurance companies that were interested in having me join them. How do you really know exactly what makes you decide when to do something?

MR. CERNANEC: Jim, your change to computer science, it sounds like you made a conscious shift from an actuarial firm.

MR. LAMSON: Sure. I had done a lot of programming throughout my career and early on in particular I wrote projection pricing programs and that sort of thing, because there wasn't any vendor supplied software. Then in the early 1980s, I shifted to the consulting environment. Frankly, I kind of agree with you that I was wanting to make a shift toward sales—if you're going to be an account executive and successful as an account executive, you need to be able to sell yourself and sell the prospect on the confidence that you're going to be able to deliver a successful consulting project or end result. Anyway, I got into that and for that purpose I found the work to be very satisfying. When I first got out of college and went to work at an insurance company, I got fascinated with the ability that the sales agents had out in the field, to be able to go out and sell death insurance to people. In other words, to go out and sell a product to people that nobody really wants to buy, because who really wants to contemplate their own death? I got interested in talking to them on the phone and later in my career I became known as a marketing-oriented or sales-oriented actuary and I was doing a lot of product development then. I think that's how the sales part was sparking me and it became a challenge to see if I could also sell. I could be an actuary and also those things. I'd proven that I could program and I could do that kind of work. I got into consulting and then for me the big leap into full-time computer science, if you will, came because of an opportunity that I recognized.

There are lots of problems and many of them don't directly involve actuarial science right now aside from the September 11 tragedies. Nonetheless, those problems create opportunities for people out there. Likewise, whenever there is a change in the insurance business, and frankly in the early 1980s there was what a lot of people considered a revolution in the insurance business with the advent of universal life, and in 1983, the adoption of something called the universal life model regulation by the NAIC. I was in the consulting environment with a large national firm at that time, and we began doing the universal life valuations for companies in support of the 818(c) deduction on their tax return and I could see that (1) there was a demand for that, and (2) companies were unlikely to continue to pay for that on a consulting basis because that's a very expensive way to get your reserves done. That's what prompted me to make the full-time leap into computer science and support through programming of the universal life model regulation of those reserves.

MR. CERNANEC: Jim, what do you see yourself doing?

MR. LAMSON: Obviously, if you look at any particular day there might only be one or two things that I'd work on that day, so I guess I'd rather answer that question as a kind of composite. My job encompasses a lot of different things, but one of those of course, is to go out and make sales presentations to these companies and that consumes a day or a day and a half when you factor travel into that, but of course, the travel part is getting a little longer all the time now. Prior to that, I have to work with the people that prepare the materials for handouts and that sort of thing.

During the course of developing the sale there's a lot of correspondence, so there's a lot of time spent literally just writing, whether it's e-mails or Word document letters, that kind of thing, and interfacing between the sales process and the technical people in our shop to be able to answer particular questions. They are questions such as, can we handle this product feature or that obscure valuation feature, that type of thing. Finally, after a sale is closed, then you have to do the grungy work of licensing and that sort of thing. You have to negotiate the license agreement and that takes a fair amount of time and additional time on the phone, but there are a lot of e-mails going back and forth and working on documents to get that done. That's the sales part of my job.

The other part of my job is to try to stay abreast of a lot of the regulatory and product changes. I also have to recognize what kinds of changes have to be made in our software and try to drive or lead that process along, and analyze the best way to work a particular new requirement into the software so that it's done in an efficient manner. We'll get into this a little bit later I'm sure, but there are a lot of careers that are related to computer science in one way or another. Computer science isn't just programming.

MR. CERNANEC: How would you describe a day of someone in your organization that's actually doing development, Jim?

MR. LAMSON: Most of the time it is of course, spent programming, but there's also a great deal of time spent in interfacing with people like myself or others in the organization, in understanding and strategizing how to best put an enhancement into the system. There's a lot of time spent at the white board where you're brainstorming how to best do these things. Often times we'll have four or five people together in a meeting and it's actually kind of a fun time during some of these brainstorming sessions.

MR. CERNANEC: Do you still use MOHAC?

MR. SMITH: You know, I think we quit using that about 18 months ago.

MR. CERNANEC: Brian, what's your day like?

MR. REID: Jim did a really good job of explaining the sales role, and since my role is about 98 percent sales, it's pretty similar; the presentation, the travel to the presentation, the preparation, the follow up, the contracts, all the same tasks and responsibilities. I don't get as involved in the software itself. I do everything I can when I have what I call spare time to learn more about it, because I think that's incredibly valuable in the sales process to know all you possibly can about the product you represent. I'm a health care and long- term care actuary, so I do try to stay abreast of what's happening in those industries. I have also been involved in the development of the product that supports those two product lines. So that's where I still get my fingers a little dirty.

MR. CERNANEC: You're still holding on to some of those packages?

MR. REID: Just a little.

MR. CERNANEC: What's a day like for a developer in your organization?

MR. REID: In our organization there are very few pure developers. There are only two. We've got a lot of the development responsibilities split among our group. You could think of a lot of people on our team as multi-skilled. We've got people who do some of the development, some of the support, some of the sales support for me when I need help in presenting or preparing something. They can assist me, but we try to mix things up so people aren't developing ten hours a day every single day. We allow them to see other facets of the business and stay a little bit diversified.

MR. CERNANEC: What does your typical day look like, Roger?

MR. SMITH: I'm not sure that I have a typical day, because I cover all the aspects. I do get involved in the sales activities of our operation. I probably spend more time, though, reviewing a lot of directions in terms of how the software should be shaped. Customers call in and they have a new product, a new feature—the pace of product development seems to be going faster than it ever has. Regulatory changes are always out there. We spend a lot of time monitoring the progress of those, trying to anticipate which ones might be coming into effect and when and what impact that will then have on the software. Occasionally, questions will come up. Which of the two technologies should we perhaps pursue? That's always interesting because we frequently have people that are very in depth with one of the two technologies, and I'm not in depth in any of the technologies really, so that's always an interesting call when we do that. I would say that a large part of what I do includes just looking at various issues that do come up and trying to decide how to implement them, because as the install base grows, it becomes more and more difficult to plan new changes without affecting a lot of other people. One person told the story about why God was able to create the world in six days and rest on the seventh and the answer is that he didn't have an installed base at the time.

MR. CERNANEC: What does a day look like for a developer in your organization?

MR. SMITH: A developer in our organization will have a series of tasks in front of him or her. They might be new features going in or they might be corrections of some prior features that had gone in, which are known as bugs, in every day speaking. The developer's day would involve trying to make sure that he or she has very clear direction, understands what the problem is, and being able to demonstrate that he or she has successfully solved the problem or corrected an error. This sounds very basic, but sometimes it can be very difficult. You might, for example, hear from a customer that there's something wrong in this valuation and you find it.

MR. CERNANEC: Other questions that folks might have about what their day might look like?

MR. MIKE WARD: This question is focused on letting some of the younger actuaries know what opportunities exist. You've explained what the days look like for the developers in your system. Who are they? Not by name, but by background and so forth. Are they actuaries? Are they students? Are they programming professionals solely? Who are they?

MR. SMITH: We definitely have a mixture of people. We have a number of actuaries that are on the development team. Some are credential actuaries writing code, writing programs for us. We also have a number of non-actuarial programmers that are programming specialists. We really do have a mixture of both. General development is part of our organization, and we a production team that mainly focuses on working with customers, but they're a very valuable and critical source of descriptions of problems or new features that are needed. It used to be very clear how to support some calculations, but I'm familiar with one that came up quite recently. We had to do a lot of work in coming up with what the process ought to be. FAS 133 was one that was a major part of our effort.

MR. LAMSON: In our shop, all our developers are all actuaries. That was a strategic decision we made early on. We thought it was easier to take good actuaries and train them to program well, so that's the approach that we've taken, with one or two exceptions on our staff. Everybody's either a student or a lettered actuary to one level or another.

MR. REID: Our situation is similar. We have one non-actuarial programmer, but she's been doing programming for actuarial systems for 25 years, so by osmosis she's certainly relatively skilled in such things, but as she's moving toward retirement, the person taking over those responsibilities is an actuary. So, like Jim, our staff is either lettered or studying, or while they no longer continue taking exams through their background, you could consider them very actuarially skilled.

MR. CERNANEC: Thank you, Brian. One of the things that I've noticed being inside organizations is that the code developed by actuaries is a little more of a maverick approach typically, yet the IT organization wants to set some things inside their enterprise architecture, a very disciplined approach. In some organizations that creates some tension. Talk about the corporal differences. Maybe I'll start with James and the decision around your developers actually being actuaries.

MR. LAMSON: I can well understand what you mean by mavericks or cowboy-type programmers, and that's a problem that organizations do have to face up to. The IT profession is a valid profession, as is the actuarial profession. I believe there is a compromise position somewhere in the middle, which is what we try to achieve when we force our actuaries that are programmers to distill, write good code, structure it in a certain way, and to be members of the team. That's another important concept I think in today's world with the state of programming today, that you don't just go out and do something by yourself. That's what the maverick programmers kind of like to do. They go off and do things according to their own style that they make up along the way. That's not the kind of an existence, I think, that you'll find if you want to work in a professional shop. It's a team effort and everybody has to play by the rules and meet the programming standards, otherwise it just turns into a disaster.

MR. SMITH: I remember one instance when I was invited out to a customer's offices and people walked in and the actuaries stood over here and the IT people sat over here and within five minutes they were hurling insults at each other. Some of them arrived a few minutes late to the meeting and I was quite stunned. I was sitting there thinking, this is with company present, how are they when they're just by themselves? I think the answer is they don't meet. There are different disciplines. Sometimes there's a tendency to assume that everyone else understands what I know and that can be a real problem. If you have somebody who has spent many years working at GAAP reporting, that person knows an awful lot about GAAP reporting, and sometimes that person can forget how much specialized power he or she has and that everybody doesn't understand how to do these things. Similarly, in IT a person can be very skilled on various issues, such as how to design databases and tune things up and can work real well, but doesn't understand why everyone else doesn't understand what the difficulties are. When it comes right down to it, you have some data that comes into a process that produces some output and if you can just keep it in those kinds of terms, it can be very, very helpful in bridging some of those gaps, but once you start speaking in acronyms and things, it can break down generally quickly.

MR. CERNANEC: We'll get to that. Brian, do you want to comment on that since in your environment, you go for the multi skilled ?

MR. REID: I don't see the tension, which is the benefit of that approach, where all of our staff members have both the IT experience and the actuarial experience. But to Jim's point about quality, we have coding standards of practice that were developed internally and are always revisited and are strictly enforced. We share those with our clients. It's an open system. They can make changes and we let them know how we approach coding the system so that things are consistent. Just a footnote—as I visit organizations, it's very much the actuarial software and then there's the enterprise software, which is everything else.

MR. SMITH: I have a theory that goes back to the early days of computing. Unfortunately, I'm not old enough to be in the early days of computing, but when I got into computing it was in the late 1960s, early 1970s. At that point in our history, I think in the United States at least, a lot of the programming was done by members of professions for their own purposes, whether that's engineers or mathematicians or what have you. When business decided that computers were a terrific way of reducing expenses and that sort of thing, they quickly zapped all the analytical people out of the world and there were still demands for more programming to go on beyond that need. So software development became a managed process at that point, and it had to be a managed process because you had people that were writing programs about things of which they were not already experts. It was not in their field and so I see us as actuaries as people who use programming quite naturally as a tool. It's something that we pick up very quickly because we're analytical and that's where your maverick programmers and so forth, I think, come from, whereas professional IT shops very much stress structure and management.

MR. REID: I know we've worked with organizations in setting up an enterprise architecture environment. Often it's not the technology that holds people back, it's actually dealing with the business processes in the organization side, the politics of what is possible around some of that. I know that a lot of time has passed since hand-print calculation tables and doing presentations. The world has changed.

MR. CERNANEC: All of our panelists are in software sales and we'll probably hear from at least one of them or two of them or all three of them that we're all in sales all the time, but I did want to take the opportunity to say that since you are in software sales, what are the critical success factors?

MR. REID: For me, because I was an actuary, I used the software products that I then went into a sales role for. I took an approach of saying I have the product knowledge, now I need to learn how to sell, and I found I was able to do that. I had people I was able to work with and they taught me some of the tricks of the trade and things to focus on and that went relatively well. I also had the opportunity to work with people who took it from a different perspective. They had a great deal of sales experience, but they weren't actuaries, they didn't understand the products that they were representing and I did not see that being nearly as successful. They made a living, but it wasn't a long-term type of thing in which they were ever going to be able to say well, I'll just learn to be an actuary. Nobody ever took that approach. They always needed product expertise with them, so they always had to have people come along on the sales calls. So for me, personally, I have been trying to blend the sales skills with the intimate knowledge of the product that I'm representing.

MR. SMITH: I would certainly second that. Fear is a major issue in selling. In going out and making a presentation for everybody, you can overcome that so easily by knowing your product real well, so I would certainly second that part of it. Being a salesperson means that you do have to be kind of a people person, but a lot of these sales techniques can be learned, and sometimes you learn them just through dumb luck and experience, but otherwise there are various materials out there for learning how to sell. The other thing, I guess I would throw into this mix is that probably the most important aspect of the sales presentation is not what you say as much as your perception in listening and understanding what the needs are of your sales process. You have to try to meet those needs, and if you don't understand what they are, you're not going to get a sale.

People other than actuaries may perform the sales job, but there are aspects to selling in which you need to identify what the need is, their ability to pay for it, and you have to learn those steps or what has to happen in order for the sale to be completed.

MR. CERNANEC: As you know the actuarial ego is pretty fragile and with my experience in sales, even when you have the best solution and have done all of the listening, there still may be a level of rejection involved. How do you cope with the rejection?

MR. SMITH: I think you just learn to cope with it over time. Back in the early days in my home office they would talk about agents making ten calls that produced three leads or something like that and wind up with one sale. Whether that's the rule or not, it doesn't really matter, but if they're making ten contacts and one sale, they're being rejected nine times. So you can't take it personally, you do have to realize that you're not going to get every sale. You might think the reasons are stupid, but it doesn't matter, you're not going to get every sale. That's just a fact of life that you have to cope with.

MR. LAMSON: I like to think that nobody ever says no; just not yet. You can take away a lot and learn a few things you might be able to work on or improve upon the next time, but yes, decisions are occasionally made that sometimes you can't quite figure out. That's okay.

MR. REID: I do like to consider myself a people person and that's the part of my job that I enjoy the most. I certainly get disappointed when I feel that I did all the right things, there's a fit and I don't get a positive decision. I always try and step back and say well, I hope I made a good impression. I hope I created some relationships within that company that could pay off down the road. There's a lot of movement in the actuarial community and more than once I've gone through a process and it hasn't worked out for various reasons, but the people I worked with ended up somewhere else and they did have a good impression and that ended up being an opportunity at a different time.

MR. CERNANEC: You are three people representing companies that sell software and there are many others. How do you keep from tripping over each other's feet?

MR. SMITH: Well, we don't. While you say there are others, there are some others, but it's a small industry and it's a small group of vendors and to a very large degree, we all know each other. In my experience it's a rare sales opportunity where you don't have competition, so you can pretty well count on the actuaries. Since actuaries are an analytical bunch, we'll often try to spreadsheet somehow. I literally had that happen where they tried to reduce the whole process to a spreadsheet, but they're going to want to line them all up. How many vendors are there? Let's identify them. That's the first phase and we help them out and make presentations typically over a short period of time so that everybody can compare themselves.

MR. REID: We do trip over each other all the time.

MR. DAVID SNELL: How do you deal with the irrational emotional counters to your sales? I'm thinking specifically of the war of languages, where you come into a company and maybe from a functionality standpoint your product meets their needs perfectly, but the IT manager or some executive who read an article in a magazine on an airplane decided that from here on in your product is no longer worthwhile because it's Microsoft-based instead of Java-based. What do you do?

MR. LAMSON: Customers were more sensitive to that five years ago. I've seen less of that today and I'm not exactly sure why. Just in terms of the pure language preference, it seems like people have gotten to the point now where there are several different languages that will work and they're not quite as sensitive to that. Now they will have other sensitivities though, maybe from a control standpoint or many other attributes. I haven't seen that, but to answer the question, what about an irrational preoccupation with one particular attribute, that may or may not be meaningful. One of the programming languages I learned in college is SNOBOL. I don't know of a company that is based on that. If you have a guy running a Macintosh or something we just don't know, I'm not sure what I can do there. That would be a very difficult situation to overcome. Sometimes you just try to overcome it. You try to argue against it or make your point, but you may or may not succeed.

MR. SMITH: The hardest thing, I believe, is in finding that out. In other words, finding out from the prospect what their emotional biases are, that either that person has or other members of the team or company have. The hardest thing is establishing a kind of rapport with your prospects where they're willing to divulge some of these things to you. You have to get them out on the table before you can ever address them and then I think you could probably divide them into two or more categories. One of the categories is things you can't do anything about. If they want your software to run on a Mac and your software doesn't run on a Mac, it's very unlikely you're going to go through all that work for one customer. So there are things you can do something about and things you can't do something about, but very often there will be a discussion that's frankly incorrect about your software. You can adjust that with the prospect and present facts, but the hard part is finding that out if that bias is there.

MR. SNELL: My follow-up question is, are any of the three of you or your companies involved to any large extent in worrying about cross- platform type developments, where you've got your programs developed now, but maybe in order to sell them in more markets you have to start making them look several different ways?

MR. SMITH: We get involved in a lot of integration activities in what we do and some of the middleware that's now been established makes some of that easier, but that's one of the reasons an organization needs to have some enterprise architecture. But as I mentioned earlier, typically you have subsystems or systems themselves considered outside of that, special tools. It's kind of interesting because there may be some opportunity if that integration or that interface was dealt with more smoothly.

MR. LAMSON: There's always the danger of spreading your organization too thin. Often you see people that might not achieve that organization's overall goal and I think there are some cases where you just have to walk away from a potential sale.

The question I try to focus on in terms of a cross platform is: Is there a great reason for doing that or is it just something that somebody wants to do? If there's a great reason for doing it, then I tend to be more supportive. We'll run in Windows and we'll also run in UNIX various aspects in terms of how we're installed. It can look very different. Sometimes after it's run on their desk, that's fine, but in other organizations we never permit that. It's got to be locked away untouched by human hands during the production cycle. I've had other people suggest getting a C compiler for their mainframe and running it on that. I tried to do that once and I would not do it again. I've had people ask me why don't you just rewrite it in COBOL with an assembler routine to handle the tough stuff on the mainframe ? Well, I'm not interested in that either.

MR. REID: In much of the process around selling and getting involved in enterprise solutions we find that there are three levels of issues—irrational, emotional, and political. I think Jim mentioned that the difficult part is surfacing the map and recognizing who represents the various levels and participation in your buyer network. Typically it's not a single person that's involved, so you have to understand what your motivations and your criteria would be for the various different representatives in that buyer network and so that's often part of the complexity of the sales process.

MR. CERNANEC: Not only from what you do from a software sales perspective, but also touching on the developer side of it, what do you see as the chief opportunity?

MR. LAMSON: Go back to the early 1980s and the adoption by New York of Regulation 126 and the so-called New York Seven, which are not seven criminals but seven scenarios for cash-flow testing. The whole aspect of cash-flow testing seemed to accelerate the development of vendor supplied software in that area, in pricing and production- type areas. I guess as I look forward toward the future, often opportunities present themselves when new regulatory requirements come through such as that one. That's not the only way, but that's one example. Should unified valuation system (UVS) ever become a reality, for example, I think there will be a lot of opportunities presented by that.

A lot of the young actuaries with programming skills and a knowledge of some of those new technologies that we're going to put on those types of products today, will have opportunities to get into computer science in a bigger way than they have in the past. I'm on the Academy work group working on the new guideline for variable annuity guaranteed living benefits, and we're developing a methodology for establishing reserves using stochastic scenario testing at the policy level. That's a first step, and I think there will be some opportunities within companies created to try to comply with new actuarial guideline dynamics if it's adopted. You need to be aware of what's going on around you and realize that you too can take advantage of some of these opportunities. It doesn't take a world leader idea to be successful at something. You just have to go out and do it.

MR. CERNANEC: So, Jim, is it the regulation or the opportunities?

MR. LAMSON: Well, certainly it's those things, but in addition to the product side as well. Look at what's been going on with variable annuity index world over the last three or four years. There are tons of problems created by writing new products with new guarantees and that sort of thing. Those problems have to be solved by somebody and that can give you an opportunity.

MR. REID: I agree with Jim. Certainly regulation creates lots of opportunities. It has for the type of software I sell and I know for the valuation world as well. We were talking about the buy versus build argument. I suspect we'll continue to lean more and more toward buy. I'm surprised it's not further along than it is, but I think if people recognize the leverage you get from a vendor package, the testing is done, the every day beating and banging on it by not just your company, but dozens or hundreds of other companies as well, that's a significant benefit. So as more companies move that way there's more room for competition, more vendors. I think as people look at getting into this sort of career, whether it's on the sales side, the development side or something in between, there will continue to be more and more opportunities, and it's just a question of, are you ready to take the risk and give it a go?

MR. SMITH: I would agree that there definitely are opportunities. Actuaries have always been willing to come up with approximations or live without some of the full analysis that they would like to do, but as more tools come online, more things become possible. We're not reaching a saturation point. We're not even close to it. As soon as you do something new, you'll have a bunch of people saying, that's great. Now, you give them the green light, but they didn't even bother asking for it because they knew it was impossible anyway. We still have that suppressed demand for mutuals, new models and additional things that people would like to do. I have a brother who is a mechanical engineer. He designs aircraft and he told me once that his engineering group spends about $100,000 per year for every member of their team in hardware and software costs.

Now, he designs fighter aircraft and things, so if they have a spectacular failure or the wings come off the plane at 50,000 feet then there will be a lot of stories in the newspapers. They don't take that chance, so they beef up and that's how they do their work. I don't know what actuaries typically tend to spend on these, but there are many modeling issues. As examples, we mentioned the living benefits, equity indexed products that just defy the traditional model. We need to get into different techniques, different approaches, and somehow make it all work and get it all done. There's definitely tremendous need and demand for additional support and different models and calculations. It's not just programs. Whether it's in a home office environment or with a professional software vendor, there are a lot of different jobs. It's just the architecture, the interface between the knowledge domain that you tend to satisfy like actuarial work and IT. There's customer support, there's a need for people who need documentation and all that—even customer support is just in the home office, because you might have multiple users of a system that you've been involved in developing and somebody needs to support those users within your company. So there are a lot of opportunities, I think.

MR. REID: One of the other areas that I see emerging, which isn't necessarily computer science per se, but it's certainly a modeling tool and actuarial skill, is in the area of valuing your customer base as an asset and moving through that around customer segmentation and being able to provide assistance around differentiated levels of service or channel segmentation. This has not been well developed yet in an insurance environment in the U.S., but it is used quite a bit in Europe and in the banking sector. As we see some convergence of industries, the customer group will be treated as an asset on the sheet instead of troublesome, which tends to be a little actuarial bias, but the customers and agents are just another opportunity.

MR. CERNANEC: Any questions or any comments?

FROM THE FLOOR: Computer science is all about technology—technology that's rapidly emerging and evolving. I've heard about critical success stuff that relates to keeping up with product knowledge and sales skills and so on. I'd like the three of you to consider or suggest which emerging technologies might an actuary pursue or become familiar with in order to be successful in computer science.

MR. LAMSON: Around our shop people are talking about the dot net development which is going on, which is VB.net and C-Sharp. Those emerging languages, I think they're in data right now, offer some opportunities for having software that can be modified by the user easily without them ever having to access source codes, for example. That would be one area that seems to hold a lot of promise for the future.

MR. SMITH: That's a tough question to answer, because it would be easier to state that the answer might be valid until you walk out the door and then it might be different even tomorrow. This is an issue, if you look back at what the hot technologies were and what you had to know over time. It would be interesting if someone had gone into the same college book stores and looked at the books on the shelves that they were teaching every three years or so going back. There was a time when APL, of course, was going to solve all the problems. There was a programming language PL/1 that was expected to take over the world.

Some of the ones Jim mentioned are brand new and have some promise. I like to think more in terms of capabilities that we need rather than specific technologies. Get a good definition of what it is that you need. For example, if the ability to allow users to modify codes somehow is clearly needed to introduce their own formulas and do their own processing, it's something that vendors have to provide. Now, how they do that—that's the tricky thing. We haven't done any of that yet and I've been told by people who have used PTS and TAS that we don't do it exactly the way they've done it. In other words, invalidating all my changes when new versions come out. There are always issues that have to be addressed. Solving that problem, getting the good definition, how much is it worth to be able to modify classes and their codes, then go on to something that will hold up and last? Thinking about technology is very difficult, very risky, because a lot of them that have flamed out.

MR. CERNANEC: What do you look for and what kind of experience do you need?

MR. LAMSON: One of the key ingredients, just like Roger was talking about, is a commitment to lifelong learning. The IT world changes so dramatically over such short periods of time that whatever the technology is, the whole world—the computer world anyway, continues to evolve and more and there are more layers of abstraction to try to handle all the emerging technologies. That requires a lot of work and a lot of reading, so one of the things that we've got in our shop while everybody's an actuary or has exams or is maybe taking exams, is that they've either decided they're going to be primarily actuaries or they're going to be primarily computer science people because it's very difficult to be expert at both.

MR. CERNANEC: I know more generally on the IT side, many companies are providing their own training. For example, my son started at a company back in May and he's actually gone through a second language course and is working on mobile technologies. He took C++ and Java in school and they just wanted to have him demonstrate that he could master languages quickly.

MR. SMITH: My comment isn't so much about the emerging technologies, because like Jim's comment, I try to stay current on the actuarial side. I know the technology that I represent, but I've definitely seen the shift in the project world, which is where I operate. In terms of the skill sets that people are looking for in companies or that vendors in this avenue are looking for, it's gravitating very slowly from APL. Ten or fifteen years ago many of the actuarial people you worked with had APL skills. I had them a long time ago, forgot them all and will never use them again, moving more towards C+ and C++. I don't know where that's going to continue, but that's really been a shift over recent years.

MR. CERNANEC: Not only the stock market performance but market capitalization goes along with that. A number of the dot-coms have been impacted and the whole technology sector has just managed to shrink. So when we say opportunities in computer science, dot-com failures, is there any relationship on how that might impact the opportunity for actuaries in computer science?

MR. SMITH: The biggest impact of the dot-com failures has been that a lot of people are asking for work that I never heard of before, including a few actuaries.

MR. CERNANEC: Are you getting a flood of applications developers?

MR. SMITH: Quite a few. The developers and sales people write incredible resumes and are the most amazing sales people in the world, but they haven't been able to hold a job more than a year at any time during the '90s. I wonder about that, but they are scrambling up some sort of ladder.

MR. CERNANEC: You each have, in your own way, been on the dark side, stayed there or moved to the dark side. I find that whenever there is some change in position or change in task, there's still something that gets discovered over time. What's different from what you're expecting and how do you interpret it?

MR. LAMSON: Well, there's a lot of management to software development. That, frankly, is something I wasn't anticipating fully. I have to say that if you go back to the mid- to late- '80s when I first got into this world, I had this naïve idea that once you develop this gee-whiz, go-faster program that you think is a world leader, that the world is somehow going to beat down your door wanting it and that's not the case. You still need salespeople to go out there and do the selling.

MR. SMITH: I would say one of the bigger things in terms of being expected is just the range of experience and expertise that different actuaries have. I made a comment earlier that you tend to assume everyone understands or knows about the same information and sure, maybe actuaries have different bodies of information that they're experts in, but it's usually within actuarial groups that different people understand different aspects of problems or product lines or functions. So it's been interesting, I'll say, just getting an appreciation for how to work through some of those issues, when maybe you have someone who is normally in charge of financial reporting that really doesn't seem to understand much about financial reporting and is navigating through those tricky waters, while at the same time hoping to sell this person something.

MR. REID: As for part of Jim's comment, I had no idea what went on behind the scenes from a technology management perspective to produce and maintain a vendor piece of software. I enjoyed that part. It was just a shock to me when I first got into it and I like being involved in it, again, even though I'm not on the secured development side of things. For me, I had an experience where I never thought I could sell anything, but when I got into it, I said well, I know the product and I believe in the product. I had a situation where that sort of stopped being the case and I had to say I can't really do this any more and move on. I was able to find another position where I could do the same sort of thing, but it was very difficult. It's very difficult when you say I'm a sales person for a product and I'm not sure that this is something I should continue to do. It's a moral dilemma, I guess you'd say.

MR. CERNANEC: In your experience, you've all worked with others that are not only dabbling, but who fully immerse themselves into some aspect of computer science. What advice do you give them?

MR. SMITH: I would recommend people to try to find a nurturing environment for themselves where their skills can be appreciated. Earlier in my career, I worked for several different home office-type companies and the orientation in some companies anyways is on a management track. That is, if you're not trying to become president of the company some day, then you're relegated to a staff position and are not really highly respected. I would encourage people to try to change divisions within their company or what have you, to try to find an environment where your skills are appreciated and you are given the opportunity to try to advance your skill sets. Also, try to find good mentors. You can learn so much from each other and someone who can take you by the hand on things can boost your career.

MR. CERNANEC: Where is a good place to work, Brian?

MR. REID: My only recommendation would be: be willing to take some risks. I mentioned it earlier, but while I was mentally going through the decisions in front of me, trying to decide whether I wanted to move into sales and so forth, I had the opportunity in front of me and was losing a great deal of sleep over whether or not to make this change. Was I willing to take the risk in changing my career path for this, and would I make it? Would it work?

I was looking at Connecticut Mutual at the time and they merged with North Mutual and there was a severance package offered and that's what it took for me to take the risk. So here I'm saying take the risk, but I wasn't really ready to do it. I don't know if I would have done it without that incentive or that sort of insurance that I could have this package and fall back on it. But in retrospect, I think about how much happier I am now and the fact that I wish I did it sooner. When you look at the actuarial marketplace, if you're thinking about trying something different, whether it's the sales end of it or the programming end, it's a great employment market right now, so the risks maybe aren't as great as you think they are.

MR. SMITH: I would just add one thought. Try to look at things as a big picture and do the activities seem to make sense. Is it something that's going to fit in long-term? I see some people that don't take that big-picture approach and try to anticipate those kinds of issues. Many of those people ended up working for dot-comers over the last couple of years.

MR. CERNANEC: Even those folks that are involved in some time spent at the screen, whether it is a laptop, desktop, or whatever type of environment, they often need to get away. I have a 12-year old son who has a number of pen pals from around the globe. This has become a little bit of a problem just because he spends a lot of time at the screen, so this is a reminder for it. I hope that you are able to get outside and enjoy life a little bit. What I want to ask before we close down is do you have any sense of nostalgia with regard to traditional actuarial views? Do you have one of those days where you wake up and say, "Hey, maybe I should be back there."

MR. REID: No.

MR. SMITH: No. I think I am traditional now.

MR. LAMSON: Quite frankly, I can't imagine going back to a home office environment for example. This is just too much fun.