RECORD, Volume 30, No. 1*_____

Spring Meeting, Anaheim, CA
May 19–May 21, 2004

Session 32PD
Data Warehousing

Track: Health, Computer Science

Moderator: CHARLES S. FUHRER
Panelists: PATRICK MCINTYRE†
                    NEIL RADEN‡

Summary: The concept of data warehousing means different things to different people. For actuaries, the primary focus is on improved access to timely information and applications that enable them to work more efficiently and produce better results.

MR. CHARLES S. FUHRER: We have two speakers today. The first speaker is Patrick McIntyre. He works for Ingenix, a health firm that is located in Minneapolis.

MR. PATRICK MCINTYRE: Let me ask a couple of questions first, to get a sense of where you're all coming from. There's a mix of health and pension actuaries. How many folks come from the health side? Most. On the health side, how many primarily work with health plans? Managed care environments? That's where I'm focusing most. How about consultants? IT professionals? None. Good, because you won't learn any IT stuff from me today. How many folks have built or assisted with building a data warehouse? There are a number of you. How many of you, although it's not a "data warehouse," actually run, retrieve and manipulate your own data today? How many of you would like to do less of that?

I come from a business background. I'm a step removed from actuarial work. I started out as an accountant years ago. I have spent most of my career in a managed care environment, primarily working with information management. For

one reason or another, I found myself gravitating toward the data. Probably not unlike a lot of you, I had started out in an organization that had bad data and I took it upon myself to try to solve some of the data problems. One project led to another, and every time I went back to the IT professionals to get some help, I found that probably the easiest way to solve my issues was to continue building on what I had started. I did that for managed care organizations over the years. Each time I tried to get the IT organization to meet my needs, and each time it became a long, drawn-out process with, from my perspective, a lack of knowledge about the business side of what we do with the data after it comes out of a box.

You're going to hear me talking a lot more about the business side of data warehouses, and not as much about "the box"—the hardware, the software. That's the easy piece. The real value in a data warehouse or in an analytic platform, which is actually what I'm going to spend more time talking about, is in whether you can actually use the data after you get it out. You spend millions of dollars on building this structure (call it what you want), build a data mart, build a data warehouse, build data cubes, buy some slick software from a company like Ingenix, spend lots of money and two or three years of your life getting all this stuff in place, then you start trying to run against the data and it doesn't work. Nothing is more frustrating than going through that whole exercise only to find out that you're going to keep doing what you've been doing because this new structure that's been put in place doesn't meet your needs. What I'm going to do is focus on the entirety of the process, the analytic process around getting data and turning it into information. I believe that the foundation of all successful information management is a solid data warehouse foundation. You have to have data warehousing, the technology support and the business connection—with the folks in this room, your clients and your constituents—to come together in a meaningful way in order for you to be able to do the job and to have the information that you need.

Are there any specific data warehousing business analytic issues that you'd like to try to get solved today before you leave here? To the extent that we can cover it while we go through our presentations, we'll try to interweave it in there.

MR. DAVID HUTCHINS: We're going through and getting business requirements. One of the things we've got planned is that we're going to have *one* definition for premium and *one* definition for member month. If there is absolutely a business need for multiple definitions, we'll do that, but we're going to limit it to two or three. Mr. Raden said that it's dumb because that's not the way people work, and he's right. People don't work that way. If you've had experience in how to build a structure that allows multiple versions of the truth without multiple reconciliations, I'd be thrilled to hear it.

MR. MCINTYRE: We'll get at that. Are there any other questions? So everybody else just wants to hear generally about data warehousing? I'm sure we're going to cover a lot of questions that you all have on your minds, stuff that you've dealt with in the past but haven't brought up. I'm going to focus on utilizing a data warehouse

to optimize business decision-making. I'm using a case study, managing medical expense trends. That's primarily what my group does within Ingenix. We build decision support solutions structures that start all the way from data warehousing and data modeling through to advanced applications and everything in between.

One of the things that I think we do really well, that to me is the core of this whole process, is the data and transforming the data and following the data all the way through the system. You just asked: at what point in time do we need two definitions of the truth? Can't everybody just use the same one? I think we all know the answer is "No." No, we can't use the same one. Not only can't we use the same definitions, but we're not even going to use the same data. But how do we do it in a way so that we don't have 15 versions that never reconcile? The real key is, how do we all have what we need to run our business but not have this huge data mess and have an ability to go back and connect back to the starting point that's consistent across all the different uses?

I started out years ago working for Aetna. I was there a couple of weeks, walked out of my office and it was as though a big light bulb went on. I suddenly realized that we're in the information management business. That's what we do. We take information, we transform it and we create premiums out of it. We track and monitor hospital contracts and provider contracts by the information that we accrue. Just about every decision point that we're dealing with in a managed care environment is all about the information. It's all about the data that we have available, or data not available that we try to then go back and recreate, to come up with business decision rules around how we run our business. There is complex data coming from all kinds of places. In claims, if you're lucky, you've got maybe two or three or four different claims systems coming to you. You've got membership. You've got encounter data. You've got financial settlements at the end of the year. If a bunch of hospitals or physician groups got paid extra money, how do you get that going back into the system and how do you break it back down again? It's very complex.

As far as location of the data, you've got claims systems, membership/eligibility systems and carve-out vendors. Not only do you have different sources of the data, but the data is in different locations. Some of the data may be on site. Some of the data may be coming from a vendor in a file. The data may be coming in all different types of formats. Dimensions and quality of data are different. You've got timing.

I give the example that actuaries are always trying to add to the data. What we're trying to do on the actuarial side of the house is take data and then fill in for all the data that's missing. On the clinical side of the house, they're trying to get rid of data. They're looking at the data and saying that a bunch of stuff is incomplete. They don't want to look at an incomplete episode; they want to look at the full truth, not a partial truth. So you've already started with a fundamental difference in the way that the data is looked at from those two organizations. From day one you're going to have differences, but what you don't need are two completely

different processes that go off and have nothing to do with each other. You'd be amazed at how often that's exactly what you have. Somebody got tired of working with IT and went out and built his own little thing. Joe Smith on the other side did the same thing. Mary Jones in some other place did something else. You come back, you try to do all this reconciliation, you spend half your time just trying to reconcile the data, and then you run out of time because now you're doing your rate filings, you're doing your financials and/or contracting with a provider. You're too late—you spent all your time on the data.

It's a very complex business. There are multiple business views: actuarial, clinical, financial, sales, operations. Every time you get those folks in the room, you've got a different source of information. They have a different statement of truth, if you will, about what the real answer is.

You've got regulatory activities that push you to do certain things in some parts of the organization. It's an ultra-competitive business environment. In setting price, there are cost and utilization drivers. We always need to figure out what's driving utilization and what's driving cost. As far as provider contracts, in my experience with five managed care organizations over the years, you couldn't even get, within any one of those organizations, people that contract the same way. So even the data coming in is different. Before it even gets into the source system, you've got it at the point of origin coming in as a per diem, versus a case rate. It's just all over the place. It's very complex. It's not surprising that every couple of years a great new idea comes out and you hear of a new organization that has done something different. It lasts for a little while, and then we're back to some of the same types of problems. We've got a very complex environment.

As far as access, we have provider contracting, networks performance and specialty services. As far as quality, there's Health Employer Data Information Set (HEDIS), care management and disease management. We're trying to get our arms around all kinds of information to make business decisions in different parts of the organization with data coming from everywhere.

All too often, information systems' shortfalls prevent effective decision making. The inability to consistently create clean, reliable trend data results in large amounts of data workarounds. Lack of ease in drilling down to the root cause of trend drivers causes excessive amounts of ad hoc data and analysis requests. Often we end up getting systems in place where somebody has captured the data and generated a report, we're looking at it and see some variances, so we want to drill through and see which provider groups that is or which benefit design that is. But the problem is that they didn't necessarily roll the data up from that perspective to be able to drill back down to us. So now you've got aggregated data. Now you go back and you ask them to go find out what the answer is. You've got a bunch of workarounds. You've got a bunch of data runs that need to be run. Now you've got a special project off on the side to answer the questions that you're generating the report to answer in the first place.

There are challenges mapping clinical to financial. I gave you the example of the actuaries that are adding to the data and the clinicians that are trying to chop off incomplete data, so you've got difficulties there. Variation in identifying and tagging of data by products, lines of business, markets, etc., all yield varying results between reporting groups. I don't know how many of you have worked in an organization where something so simple as defining a market is not so simple if you're working for a multi-site or a multi- regioned organization. What New York is to one person is something completely different to another. Yet that's the routine. That's the norm; that's not abnormal.

All the above, combined, result in a lengthy analysis cycle with slow turnaround times for reporting and analysis. It's no wonder that you get people like you, like myself, who have a need to get to a certain decision and who get the data on our own and solve for it on our own, to the level that we're comfortable that the data is good. Yet, often the person who has the "good" data has missed something critical and is actually making decisions on bad information. The person has incomplete or inaccurate data, never even knows it, and keeps staying on his or her path. We never go back and work with the other organizations to try to bring the data issues together to solve for them in a universal way.

There are key issues to overcome in effectively managing medical expenses. I'm not going to go into a lot of detail. Effective trend management programs invariably cross functional areas. We're going to contract with providers out of one part of the organization and we're going to run our disease management and care management programs out of another part of the organization. We're going to set premiums out of a third. The entire organization is split into different kinds of decision-making pods, if you will, that are trying to solve for different things.

Benefit design strategies are extremely important in directing cost and utilization. Proactive underwriting can make a huge difference between operating gains and losses. There is probably nothing earth-shattering there.

Focused provider contracting initiatives and effective network management can have huge impacts on medical cost containment. Medical management efforts need to be carefully targeted and constantly evaluated. Claims operations need to be able to support the organization's benefit design, provider contracting and care management strategies.

How does that translate into the data problems? You have the trend issues cutting across the multiple functional areas. Data integrity, as I mentioned, is an issue. You've got disparate data sources coming from multiple business analytic platforms. At the root of the tool, the report is a set of definitions that somebody created based on who knows what. They could have been his or her own concoction of what an appropriate set of definitions is. They could have been something that they were trying to benchmark against, such as Milliman & Robertson benchmarks.

Benefit design strategies are effective in controlling cost and utilization, yet benefit impact analysis is complex and data intensive. That is something that most of you on the actuarial side find pretty simple. You probably deal with that on a regular basis, but often those are solved for in one part of the organization, and yet you'd be surprised at how many clinicians are over in the care management organization creating their own benefit design definitions because they grabbed a field in a data set, thought they knew how to use it, and started creating benefit design analyses.

Data structure issues often inhibit optimal analysis. Encounter data is often absent. When it comes in, it's generally missing, incomplete or applied incorrectly. Out-of-network and out-of-area data are often lacking. In underwriting, demographic trend information needs to be accurate and reliable. Information on developing medical technology is a solid, predictive lever, yet difficult to incorporate at a detailed data level. Often the forecasting tools that I've seen that apply, the new emerging medical technologies, do it at the very end of the process and look at it from some kind of futuristic standpoint forecasting capability. But often those findings and the projections that the actuaries are working on don't get brought over to the clinicians to look at when they're managing the different care management programs.

Some of it is simple sharing of data, but as you see, we're going to be building toward how we get all this in one place so that we don't have to deal with multiple, disparate processes throughout the organization.

Focused provider contracting initiatives and network management have a huge impact on medical cost containment, yet the data structures need to be able to support a variety of provider contracting strategies. Again, in all the organizations I've been in, even the data capture side is not consistent within an organization. How a claim actually gets put in the system and listed, whether it was a per-diem contract or a case-rate contract, is important for being able to get that data out later in a consistent manner.

Cost variations between like-kind providers can provide a wealth of information if they can be effectively reported on.

Medical management activities are often managed among different departments with different IT platforms and different results. Even disease management programs are often subbed out to multiple vendors. The data coming back is often very difficult to get your arms around. ROIs cannot be precisely determined for many medical management programs. Claims operations need to be able to support the benefit design, provider contract and care management strategies. Medical management initiatives will only provide results if they can be implemented within a claims adjudication system.

Claims payment activities are the genesis for all medical cost management programs, so data integrity is essential. Effectively managing medical trends

requires a solid, stable data foundation. What I have experienced over time, in working with organizations, is that if you're going to get your arms around data in a meaningful way, it has to be built on. There has to be a starting foundation, an enterprise data warehouse of some sort, where you at least bring the data in to one single storage location, and at that point in time, that is where you're doing your basic low-level data cleansing. You're getting rid of duplicate claims. You're bringing in settlements and figuring out what to do with them. That's filling in the completeness question, if you will, by getting all of that data in one place and starting to create the definitions that are consistent throughout the organization. What is a medical day? What is a surgical day? What is a cardiologist? How do we roll all those definitions together?

Layering on top of that, you're going to want to start looking at the specific business needs. So building up from the base foundation, which again is where the data warehouse becomes essential, you want to be able to, again, solve for all of the data formation, the data cleansing, at a level where it's consistent across the entire organization. If there is anything that can be done with the data, that everybody is going to benefit from, I would do it in the data warehouse. That's where I'd start that initial level of cleansing.

The data mart level is where you're starting to get to the point that this gentleman had asked about. Now where do we look to get the multiple business uses that we need to run the business? At the next level up, you're choosing which data from the warehouse that you're going to use. The actuaries are going to use most of it and then add to it. The clinicians are going to pull from the data warehouse only those episodes of care that are complete, or that meet their criteria for being able to look at medical conditions. On top of that, you're going to want to layer in the data cubes, or whatever technology that you're going to use, to be able to access the data in a meaningful way and to standardize the reporting that you want to get out of it. Then layer on that the different applications and EIS. But all of that is first predicated on having a business focus on running the organization. What is it that you need from an underwriting and actuarial perspective? From a claims operations perspective? What do you need in order to run your business? Define those business requirements that you need to effectively run the business.

Then you're going to want to cut through the organization, cut through the hierarchy of technology platforms, to make sure that the data warehouse and all the other technology platforms sitting on top of that are able to deliver what you want to get delivered.

In the typical information management and/or infrastructure approach, each area of the company gathers its own data using its own assumptions. Again, you've got the actuarial/financial folks looking at it a certain way, having their own systems, capturing data from the claims systems, from the member enrollment systems and from pharmacy and labs. You've got clinicians going out and doing it a different way for a different set of purposes. You naturally end up having different data stores

across the organization that create all the reconciliations that you're trying to get away from. That also ends up creating many ad hoc data requests because you're invariably going to be asked questions from one side of the organization to another.

One of the organizations that we've been working closely with for the last couple of years has a continual battle around the actuaries bringing us the trend information, so we see costs doing this and we see utilization doing this, and they go back to the network management group and the network management group says that they're wrong. The network management group says, "We've got our contracts here, we've looked at our contracts and here's what the contracts are telling us. Here's what the trend on our actual contracts is doing." On the other side, you've got the care management folks saying, "We've gotten live data. We know what our census is. We know who is in the hospital. You've got the actuaries over there *guessing* at what the runout is. That information is not correct. We've got all the data sitting out here now on exactly who is in the hospital." You've got three different sources of truth.

We've been working with them to bring all three of those different organizations into one platform. It starts with the data warehouse and forcing all of their data into one data warehouse. So far, we've gotten all the provider contracting information that they do for pricing to come out of the trend data warehouse as the standard, and we've got all the pricing now on the network management side coming out of that warehouse. We don't have any issues anymore about whether it's a medical day trend that's higher or a surgical day trend that's higher, and the two of them arguing over which one it is, because they're now using the same data at that level of definition. There is only one medical day or there is only one surgical day number for a given calculation. In the last several months we've been working with the clinical group to get all of its business definitions, and we're looking and doing an analysis across the now combined health plan data mart, which is the trend in the physician/provider management side. We're now in the process of getting all the business requirements mapped out against what's currently in the data warehouse so that we can add whatever additional fields we need. It's a long, painstaking effort, but at the end of the day it pays off and you end up having a significant amount of resource efficiencies. On the network management trend integration— huge organization— across many plans, out in the field, they ended up saving 25 ad-hoc data analysts who they were able to redeploy onto other things because they simply didn't need to do all the reconciliations that they had to do before.

All of this obviously results in inconsistent analysis and reporting, wasted time on excessive reconciliation and potentially conflicting business decisions.

Creating a common data language and process becomes a necessity. Common data foundation and data acquisition methods are essential. Again, you want to get all of the data that you can possibly get in one place. If there is no reason for somebody to have a piece of data off in another location, you want to force that back in. If it's something completely obscure that doesn't fit the needs of any of the other

members of the organization, there may be some reason, but I haven't run into too many of those instances.

You must have consistent business rules, including naming conventions and definitions. Again, you've got to force some parts of the organization to agree to a new set of definitions. If medical days and surgical days are driven primarily by the actuaries who are doing benchmarking with a number of benchmarks available, try to work with the contracting folks to be able to accept that. Can we actually change the way that they do their contracting to be able to better allow for that matching? It sounds like we're talking about having business folks change the way they do business to meet the data needs. To a certain degree, you're going to have to do that on occasion. If you want to end up spending more of your time analyzing the data and less of your time chasing after it, you're going to have to force the consistencies and terminologies.

Again, in some instances it's not an easy process, but if you stick with it, you end up getting to a point where you have people talking the same language, with the same business rules and the same definitions. You'd be amazed at the improvement in communications.

Obviously, once you solve for all the business rules and requirements, the coding and the programming come in. As far as overlapping and/or integrated health data models, again, once you've identified the data that you're going to use, you have the business use for it, then the clinicians are going to pull that data. They're going to go do something else with it. They're going to have ETG groupings, if they're going to use that. They're going to have completion factors on the actuarial side. But when you actually go back and do a reconciliation, you don't have to do a full reconciliation. You're actually just working back into the data to find out at that lower level where you both started from. It makes the mapping back to the source data so much easier. You can actually roll back down, do an analysis and say, "Yes, we started here, you started here and we got the same data. We're ready to go. Let's use your data for your purposes and let's use ours for our purposes and know that we're making the right business decision. We're not using two different sets of data."

For modular, interlocking applications aimed at enterprise-wide strategies and solutions, having consistent data, data definitions and mappings allows you to use applications consistently across the organization. One organization has used Cognos data cubes. Once they built their data mart and had consistent definitions in there, all of the programs now that they keep coming up with that they want to build new reporting capabilities onto, they actually keep coming back to those cubes that we built for them, and they keep adding on new programs. They started out with medical trend reporting. They came back and said, "We want to do ancillary provider contracting through this data mart with these data cubes. Can you do that?" Yes, we did that. We built those for them. Then they came back and said, "We want to do facility pricing off of this. Can we do that?" We went back and

looked at the definitions. We had brought them into a consistent set of definitions that matched what the actuaries were using, and it was also consistent with the way they were contracting. Yes, we can do that. So now they do all of their large-scale hospital contracting off of the same data mart, the same data set.

They started a Center of Excellence program, where they're looking at cardiology and oncology programs. They're looking to target specific hospitals and provider groups within certain markets across the country. Are we able to do that off of this data mart? The answer is yes. So they keep coming back to us. In the past, before we built this for them, our response would be "Sure we can do that. That's about a $600,000 or $800,000 effort." Their eyes would gloss over. They come back now and we say, "This is an extension of the system that we built for you. We have a pool of dollars that's part of the service we provide. You can use it however you want." Now they, within their organization, prioritize whether they want to build more ancillary reporting capabilities or provider contracting capabilities. But it's consistent; everybody uses the same tool. It's built off of the same data set. There are the same definitions. They're able to continually go back and build more analytic capabilities off of that same platform.

Leveraging a single common analytic platform is the optimal approach. You've aligned the organization across the organization to accept the claims data and member data into a staging area. You've gone in and you've agreed on common conventions, common business definitions and common metrics. You've brought that into a common data foundation, which, again, often is that data warehouse or data mart, depending on what your structure in your organization is. Now you have the different parts of the organization that are actually feeding all of their data from the one source.

An appropriately designed and implemented data warehouse is the answer. You basically have a hierarchy of technologies that are going up the organization. You've got source data systems at the bottom of the pyramid. You've got a data warehouse, which, again, is the foundation for everything that you're going to do, on top of it. If you haven't built a consistent integrated data warehouse, anything else that you're doing upstream is still going to be a struggle to reconcile back and to force people back to the same definitions.

Sitting on top of the warehouse, you then start breaking out data marts that allow you to specialize the data for use in specific parts of the organization. Sitting on top of that you've got data cubes and executive-level analytic capabilities. Folks are using this data set and talking about strategic business initiatives, corporate-level initiatives and business unit programs. There is action planning and action plan implementation. On the other side, when you've integrated, you have the IT folks, instead of trying to sell you on a warehouse or a mart or a tool, now aligned where they're talking about those things, but they're doing it in a way where it's tying back to the business functionality that you're trying to achieve. What you have are lines that are cut down the pyramid that are based on the business use. You still

have a financial group of folks or an actuarial group of folks who have fine-tuned the data from start to finish for their specific purpose. You've also got another sliver that allows the clinicians to build their programs through the entirety of the pyramid. You've also done that in a couple of other areas. But what you've done is, at the data warehouse level, when you do that drill down, you've got just one data warehouse. You don't have them drilling back to their bucket and their bucket and their bucket. You have them drilling back to the same data warehouse with consistent information.

An example of an integrated analytic platform built-up foundational data warehouse is Ingenix Trend Management Solutions, products and services providing medical expense information and solutions across the information chain. We start with a data warehouse, "Mindset" is the product name. We build from there with data modeling for cost and utilization, specific purposes. The Trend Analytic Data Model (TADM) is a very specific data model that we use for the actuarial community. Off of that we build our reporting, our health-care trend analysis reporting, our Procise clinical decision support tool. We then have trend analytics that are offered through our consulting group Reden & Anders, and then we have trend forecasting tools. At the end of the day hopefully you'll have profitable decisions. All that starts with *one* warehouse that goes through a couple of discreet sets of data models that are then built into everything we do. We run about $60-$70 billion worth of health-care costs that we manage through this process consistently across all of our clients.

It has taken us about three years in one particular place because we had so many moving parts to the organization, so many mergers and all. We stuck with the plan, and you know what? We're going to have one warehouse; we're going to have one source of truth. We're going to allow people to customize for their specific business needs, but only when the data has gotten to a point where it makes sense for them to break it off and use it for their purposes.

If it's capable of being done at one level, we force it to that level. Anything that gets done at a higher level of the pyramid that we figure out there's better business value for the other parts of the organization, we grab that and force it back down in through the pyramid to the lowest level in the organization that we possibly can.

It takes time. It takes commitment. At the end of the day, once you've made a couple of major incremental improvements in the way you run business, you start seeing people embracing that. People start asking, "Hey, wait a second, can't we do such-and-such in the data warehouse? Why are we doing that over here?" We've seen people come back and volunteer to help figure out how to get our data into "the warehouse."

It's pretty powerful stuff once you get into it. This talk was more about the need for it and the concept. Neil is going to talk more about the specifics of data warehousing.

MR. FUHRER: Neil Raden works for his own firm, Hired Brains, Inc.. The company is located in Santa Barbara, Calif. Neil has spoken before at the Society meetings.

MR. NEIL RADEN: I thank you for inviting me. I really do like coming and speaking to the Society of Actuaries. I am plugged into this whole gestalt of being an actuary, having been one for a while and having suffered through some exams, but that was a long, long time ago.

What I'd like to talk about today is how we built data warehouses for actuaries, in particular, in life and health companies. I'll talk about what was involved, what was the point of it and what was the result. I'm going to sprinkle that with editorializing about the industry. Data warehousing has been a huge boost to a lot of actuarial departments that I've worked with. It's also been a big failure for some actuarial departments that I've worked with and, presumably, for some that I haven't worked with. I'm going to discuss briefly some of the reasons for some of those failures. Feel free to interrupt me with questions.

Hired Brains was a magazine that I started for consultants a number of years ago that flopped. So I changed the name of my company as an homage to this huge investment I made that didn't go anywhere. We're consultants, typically. We work with companies like yours building data warehouses, business intelligence, analytical systems and so forth, predominantly. Personally, I spend a great deal of time consulting to the software industry. I help them with their products, strategy and getting the marketing messages out as well. I do still spend a great deal of time doing this kind of work.

I want to talk about a timeline and what I was doing in this period. This is hardly a comprehensive history of the industry, but I want to point out where we've been and what we've been doing. Back there in pre-history in 1975, before there were computers, is when I worked as an actuary. We went through host-based decision support and executive information systems in the 1980s. Thankfully I skipped over the PC spreadsheet part and went straight into client/server applications in the mid-1980s.

We picked up data warehousing about 1990. I've been very vocal in data warehousing all of that time. I write quite a bit. I speak. I've been maybe even outspoken about it. I think that some time in the next five years, data warehousing is going to peter out. That's actually good news, and I'll tell you why. What follows is analytics, and that's a little limiting, but I'll try to explain what that means.

Data warehousing as we know it has been around since 1985. As I'm sure you're aware, things have changed a little since 1985. We did have Moore's Law, but not the Internet, the worldwide Web, or a hundred other things. However, many of the basic concepts of data warehousing and methodologies still prevail. It's kind of amazing. They shouldn't. Things are changing.

Here's the problem. Someone asked about multiple versions of the truth. That's kind of an alarming concept, and I'm not surprised you're alarmed by it. Meeting statutory needs is a perfect example. What do financial actuaries do who have to meet statutory needs? They go through a great deal of number crunching and a great deal of analysis for reporting purposes, typically for things like solvency and asset/liability modeling.

That's not particularly useful for understanding how well your company is doing, what the problems are on the horizon or how to solve problems. Those are strictly statutory requirements. Right there is an example of two versions of the truth. There is the statutory version, and there's the business performance management version. We may be dealing with the same data, and at the very lowest level we may want to have a common definition for that data, but how we deal with it in different domains could be very different. Those distinctions could be very subtle or they could be very big.

When I say that it's tricky to decompose profit so you and your stockholders can understand your business, let me give you an example of a bank that I was working with not too long ago. The bank decided to incent all of its loan officers for implementing a new customer resource management (CRM) system.

They were going to get closer to the customers. The first thing they did to get closer to the customers—this was commercial loans—was eliminate the ability of the commercial customers to actually call him on the telephone. They now had to either use the Internet or dial into an interactive voice response (IVR) system where they talked to a computer. Of course, the bank's customers hated it. Which customers fled first? The bank's most profitable ones.

What was happening? As the business was deteriorating, they were releasing reserves, but the reserves were based on an average basis of the book of business. Which business were they losing? They were losing the best business. Eventually their reserves had decreased to the point where they were insufficient, and they had to be bolstered. But they had been getting bonuses for eight straight quarters because their profits kept rising. That's a perfect example of using the numbers to calculate the statutory profit, but not understanding your business.

There are all sorts of problems. When we talk with actuaries about what data we need to do our work, they look at me and say, "All of it." They're right. Unlike other parts of the company, it's true. How do you do a valuation without having access to everything? You have to have everything. You have problems that are unlike other parts of your company and other kinds of companies. Data warehousing wasn't even invented for insurance companies. It was invented for consumer products, manufacturing and companies that have relatively simple and flat models. Your models are just out of control. Your data sources are a disaster. The IT organizations in most insurance companies I've worked with are not cutting edge. They haven't done the best job in the world they're working with. It's not really

their fault, because they've been underfunded and they're working with terrible old software and terrible, old hardware. That makes building something for an actuary very difficult and very expensive, because the hardest part of this is assembling the data, as Patrick said. It's like making sausage or violin strings. You like the result but you really don't want to know what goes into it. That's where are.

What you end up having to have is a whole data warehouse. You can't build a little data mart for the actuaries; you have to have the whole megillah. We've got all of these profit centers, cost centers, data definitions, data sources and data flows—I call it the "uber" data warehouse.

How do you go about doing that? Someone asked me this morning, "We've got an IT organization that is doing some sort of enterprise thing here and some sort of standards thing here. How do we get their attention? How do we get them to help us do this?" My suggestion was, "Don't tell them what you're doing. Don't call it a data warehouse. Call it anything else. Do it on the sly. Start small. Do a small server. Find some way to do it."

Frankly, my experience with actuarial departments is that you have a tremendous amount of power and a tremendous amount of ability to get things funded if you believe in it. The most important thing is that you have a vision of what you want to do. What is it going to be like when you're finished? You don't want to just say that you have to be able to generate reports faster. Nobody is going to fund that. Don't say that the work is too grueling, you have to write all these COBOL programs and you want a better way of doing that. No one is going to fund that. You should say, "We're really going to change the way the actuarial department works. We're going to take the people in this department and we're going to allow them to work up to their ability and their training instead of doing a lot of menial stuff they have to do now to get their work done." People will fund that. They love to hear an idea like that. If you start small and don't threaten the powers that be, you can usually get started. Once in awhile, they'll even join in.

Actuaries see a data warehouse as statutory reporting, reserves movement analysis, experience and so forth. People in IT see it as a big technology piece. You've got to find a way to bridge this gap. The problem is that when people build data warehouses and IT is involved, they usually start with the real big machine, because they like to buy one. Then they want to buy and build a real big database. The first thing they do is go out and talk to a hundred software vendors. They want to buy front-end software and ETL software and database software and data profiling software. They have this big technology initiative, but nobody has talked about reserves, valuation or statutory reporting. You have to drive the project yourselves. You have to keep the focus. It's easy to say but it is very difficult to do. It's all in the way you organize the project and get things started.

Frankly, the best thing to do is not call it a project. Projects have a beginning and an end. This is never really going to finish because you're not really doing a project.

You're changing the way your organization operates. The way to keep this focus is to start with the benefits. How good is it going to feel when we get to the end and this actually works? Then you work backward from there. Some people like to start with the pain. What's the pain that's going to go away? It's the same thing, isn't it? I'm going to go through a little case study here in a minute and explain how all these things actually happened in a project. But first, I want to launch a polemic against the conventional wisdom of data warehousing.

Current data warehousing is the enemy. It's a relic. It's something that is absolutely not capable of supporting modern business intelligence. This is the corporate information factory designed by Bill Inman and, later, Claudia Imhoff that is the de facto standard in enterprise data warehousing. There are a zillion pieces of technology but there are no people in it. The users are nowhere. The focus is all wrong. A good data warehouse project diagram should be filled with people. Then you'll have a successful project.

Buried in a conventional data warehouse diagram are things called workbench, library and toolbox service management. Go through the literature and try to find the description of what that stuff means and how to implement it. You won't. It's just a diagram. It doesn't explain anything. You see flows of information on the diagram, but that isn't really how information flows. Things are labeled like "DSI," which stands for decision support interface. What is that? Those things are all relational databases. They all have their own physical data model. Each one of them has a different interpretation of the data and they call that the single version of the truth. That's doublespeak if there ever was doublespeak. That's pure propaganda. There is no single version of the truth in that mess. That's the end of my editorial.

All of the physical aspects of data warehousing need to disappear under the covers. There's going to be a layer of controlling software around it. These things are all coming out now, and some are already out on the horizon. When you build a model, you're going to be building a model in purely conceptual terms. When you talk about an insured, a coverage, a policy or a group, those will be terms that you can define here. You can have as many definitions for those things as you want because definitions are multidimensional. To IT people, they're one dimensional. The definition of a customer is X. What's the definition of a marketing analyst? It's a person who analyzes stuff for marketing. What's the definition of a marketing analyst when there's a fire drill and the marketing analyst is the fire marshal? It's a different definition, but it's still the marketing analyst. What's the definition of a customer? If you're a salesperson, a customer is anybody in the world you can get in front of. If you're in finance, a customer is somebody who owes you money.

They are all customers. The point is, if you take those names and you think about the fact that they exist in more than one dimension, there are potentially different definitions for them and different instances for them. They're sparse. There may not be a definition for them in every tube in the grid, but in some of them there will

be.

That's what I mean by multiple versions of the truth. It is impossible to do that with current data warehousing technology, except by doing what I showed you in the enemy, by building layer upon layer upon layer of data marts and cubes and other pieces. This all becomes a giant maintenance problem because if we ever have to change anything, we have to first find out all the zillions of things that are affected by it. Then it takes forever to change them.

I'm going to talk at the end about a survey we did about how people feel about their data warehouses. It isn't pretty, and a lot of it is because technology people dream up solutions in their heads and if the solutions seem elegant to them, then they assume it's the truth. They don't take any cognitive view about how people work, but people's work is incredibly complex. What we need to do is solve problems for people on an end-to-end basis, not just implement some sort of a solution, and say, "Here, this is so much better than what you had before." That isn't adequate. It just isn't going to work.

The case study is from a Canadian company, but the company did business in the United States too. It's a life and health insurance company, so maybe you'll recognize some of these names like Vantage, Capsil, CFM and CAMRA. These were the kinds of source systems and this is where we started. Each one of them had multiple extracts that, of course, the actuaries had to build. So most of the actuaries were programmers in delightful technologies like COBAL, QMF, FOCUS and APL. Of course, there were multiple extracts from each of those systems. But this was how they did their work. In this particular case, the regulators came to them and said that they had to do valuations on a quarterly basis now. This being Canada, they were seriatim valuations. You know that's a big mess. They said, "That's okay, it only takes us five months to extract the data. Wait a minute. We can't do four of those a year!" So this was their problem. Talk about pain.

At a meeting like this, I think it was a spring meeting in Washington, DC, and in 1996 or 1997, the chief actuary came up to me and said, "This is the problem I have. Can you help me?" I said that I thought I could. This man was unbelievable! He knew exactly what he needed to do. He knew exactly how he was going to do it. He knew how to slick the IT department and do an end run around them. He knew how to get the appointed actuary to fund it. He had 22 or 23 actuaries working for him and they just loved him, because they knew he was going to make things better. They believed in him. It was a true visionary thing and that's what fueled this whole project. But I have to tell you, there were times when we took the server, locked it in his office and put people on the project doing guard duty 24 hours a day, so IT wouldn't come and take it. I'm not kidding! But in any case, it looked like a spider web, with all the different extracts. It's not just a one-to-one situation, and this is mild. In this case, we had 24 or 27 different source systems. All of them had many extracts into many different reporting systems and subsets. It was a nightmare. This was what we had to deal with, and this is why it took them

five months to do a valuation.

Here was our original plan. They were going to use a tool called AXIS that was sold by GGY in Toronto, which is actually much more than a valuation tool, but we used it strictly as a valuation engine. All we had to do was feed the raw data to AXIS by line of business and so forth. It would run the valuations and then send all the cash flows and the rest of the data back to the data warehouse. Actually, I'm ahead of myself. That wasn't our original plan. Our original plan was just to build a detailed data warehouse of all the necessary data for valuations and feed the AXIS tool. At that point we really weren't funded to do anything else. Then they would extract the data from AXIS and use spreadsheets to do the rest of their work.

It didn't take long for us to say, "But now wait a minute. AXIS gets this much data in and it sends this much data out after it's finished with its calculations. Why don't we look at AXIS as both a target and a source of the data warehouse?" That is what we did. We pumped all of the different runs, and when we did a second scenario or a tenth scenario, we didn't delete the scenario runs from the data warehouse, we just tagged them with a scenario ID and just kept building the data warehouse out. If we wanted to go back to an AXIS run from three weeks ago, we could compare that data. We didn't have to throw it away. We already had modeled it that way.

Then we decided that since we had all of this data in the data warehouse now, not only the detailed information to perform the valuation but the results of the valuations, we would build analytical models to do all of these different kinds of reports and analysis that we needed. We also did the United States, by the way. We did U.S. statutory and Canadian statutory, including New York Commissioners and all sorts of specialized reports. We were able to generate all of this from this single data warehouse. But we didn't stop there. The company—and this is where IT finally came in—was spending a trillion dollars on implementing CRM. They sent out a request for proposal (RFP), and some consultants came in and said that for $5 million, they could extract the data. We said, "Guess what? We already have it. We have all your customers. We have all the sales data. We have everything." It was an easy win. We saved the company a fortune, because we'd already done that. That put the actuarial data warehouse on the map.

Now of course at that point, IT took over the server, moved it down to the data center, put a database administrator (DBA) on it, added their overhead and so forth, but with service level agreements that we had negotiated with them so they didn't mess it up. It actually worked pretty well. That was an easy win. It paid back the cost of the system 300 percent in the first six months and what they "saved" by hiring a consulting company to build it.

It wasn't the CRM system. It was the data to do the analytics that related to policyholders and insureds. We did a couple of other things too in the second phase. Once we had all that up and running, we chose analytical tools that, instead of generating a bunch of reports or generating a bunch of cubes, allowed the

actuaries to build what we call "reusable analytical applications and components." They used tools that got incrementally better. If you had a routine to do movement analysis, for example, then you could put that in the library for the actuaries. Anyone who wanted to do movement analysis would grab that component and build whatever other pieces of programs around it to do additional analysis.

As far as the technology we used, we stumbled on that. We started with a product that had huge promise and turned out to be a big failure. We ended up using a tool that is still in business but not exactly prospering. We eventually switched to Microsoft Analysis Services and a couple of different front-end tools like Panorama and ProClarity. It turned out to be hugely successful. The actuaries loved it because it took a minimal amount of time for them. The most important thing for someone who is analytical is to be able to use a tool that's expressive. That means I can go to the tool because I have an idea in my head and I can get it into the tool without having to jump through hoops. That's why people love spreadsheets, isn't it? We all know that they're expensive, not collaborative, not quality and they create a maintenance problem. But they're expressive. You have a model in your head, you go to a spreadsheet and you can make it work.

What you want is a tool that is expressive like a spreadsheet, actually more expressive than a spreadsheet, but has all those other qualities, meaning that it's maintainable, it's manageable, it works in a collaborative workflow environment and it's secure. That's the important thing.

By the way, we did make the quarterly valuations and we did it with six months to spare, which was nice. We used the tool to develop experience studies, movement analysis, all the asset performance and all these other kinds of things. Now it has been six years since this has actually started. Let me tell you what has changed in that company. Where turnover used to be 50 percent or 60 percent a year among the actuaries, it dropped to almost zero. Now like all good things, it has crept back up again a little. But people like coming to work now because they actually have something to do. They have something to do that they were trained for and that they're interested in instead of writing COBOL programs. The benefit of that is immeasurable. It goes well beyond measuring something like the ROI or the payback of the data warehouse.

Think about the amount of time that it used to take to hire an actuary, bring the actuary into the company and teach the actuary where all the data was, where all the programs were, and where all the JCL was. The ramp-up time to get somebody new into the organization was tremendous. Add to that a 50 percent turnover and no wonder they couldn't get any valuations out. But think about what they can do now. It's been a huge morale booster in the department. It was a real delight to come in and see, where before there were 20 people in cubicles sitting there with their heads down banging on keyboards or reading manuals, people now gathered around the desk, three or four people at a time, solving a problem. We no longer had the annuity actuary and the health actuary and the traditional life actuary all

working on their separate projects. They would come together on an ad-hoc basis to solve problems, and then they'd form other groups to solve other problems. It was fantastic.

How do you do it? I'll give a simplified version of a project team. I call this acquisition, storage and access. These are generic terms, and people's roles in a data warehouse project aren't this constrained. But the point I'm trying to get at is that each of those three functions is really not the role of one person or a couple of people. People with different capabilities and from different areas have to jump into those processes and work collaboratively.

What do we have on the access side, the part that faces users, including the analytical models, the business intelligence tools and so forth? We have the principal consultant working there. We have the program manager working there. The application developer and subject matter experts work there.

The same happens in the middle, which is the storage piece. Of course, the DBA has the biggest part of that, because storage is a DBA function. But the architect is involved in that, the application development is involved in that and even the subject matter experts have to be involved in that. The whole thing has to rest on a foundation of infrastructure support and a project manager to keep the whole thing going, but the program manager is the glue that holds everything together. That's the visionary. That's the person who keeps the project going, who provides air cover, who gets funding, who keeps the different factions here from squabbling and settles disputes. The most important thing is that the program manager has that vision thing. The program manager understands why he or she is there and what he or she is doing. If you don't have that component, you might as well not even start. You might as well just go spend the money on something else because it isn't going to work.

The vision is the most important thing. The technology is the least important thing. Methodology is next least important because the best methodology in the world doesn't help you if you don't have vision.

How do you do it? There are always going to be priorities. You're always going to have to decide what to do next or what things get pushed back or how much time you're going to spend on things. You have to have some kind of metric, but remember, whatever metric you come up with, it's a model. Like Deming said, "All models are wrong. Some are useful." Whatever measure you come up with is only going to be a proxy of what you're trying to figure out. But you have to find a way of prioritizing things. Always start with results. What is our priority, starting from 1 to n? All those results are going to take you through what analytics you need. They're going to expose to you what the models need to be to generate those analytics. That's going to get you talking about definitions and data. If you look at traditional data warehousing methodology, they all start with asking, where's the data and what does it mean? That is absolutely the wrong way to run a project.

You're going to run right into the ground if you do that. Always start with the results. This is the business focus. Business focus can be something very prosaic or it can be something very lofty, but it has to be a result that you're aiming for. It has to be measurable. That's going to guide you because you're going to face a zillion decisions you have to make. There will be a lot of forks in the road.

Last year, as a prelude to going out and pitching a multi-client survey to a number of software vendors (the topic being, does business intelligence improve a company?), we conducted a mini-survey of our own of just seven questions. We went out to 300 people in 66 companies. The criteria for being part of the survey were that you had to provide at least four users of business intelligence software and no more than one IT person, because we had specific questions for IT about budget and architecture, but we really didn't want to talk to IT people. If you look closely at most surveys in data warehousing—the Data Warehousing Institute is notorious for this—you'll find that 70 percent of the people they talk to are consultants, vendors and IT people. They are not people who have actually used the software. That's why the results we're getting here are so vastly different. I've got to tell you something. I've gotten hate mail from an article I wrote in *Intelligent Enterprise Magazine* like you wouldn't believe. Everybody in the data warehousing industry has dug in their heels and they're full shields up over this.

But it's 300 people. Does that mean this is the whole industry? Maybe it was a bad survey. But we've blown it out now to 100 questions for the multi-client survey and we haven't compiled all of our statistics yet, but the answers we're getting are not very different. Here's what they say. After the data warehouse is built and all the information is rolled out, for a marketing analyst, how many hours a week do you spend using your business intelligence tool? Practically nothing. Data administration is what I call the "data jerk" work. You're just mushing files around, sending files, going and looking for data and that sort of thing. This is after companies have spent millions of dollars for this, and this is what they're still spending it on. This is a disaster.

The good news is that there's a way to predict what works and what doesn't. If companies spend less than 1 percent of the total budget on training, exactly 0 percent of the people surveyed think that the data warehouse effort was worth it. But if they spend 20 percent or more, 100 percent of them do. Training has nothing to do with this answer. It doesn't really matter what the companies spend. The correlation between what the companies spent and how people looked at the effectiveness of the training approached zero. Think about that. Of the companies who spent 20 percent of the budget on training, some got good results and some got bad results. But it still had a positive response on the quality of the effort.

The reason is that those companies that spent more on training were precisely those companies that were focused on the goal, not on the technology. They were focused on doing something. They were the kinds of companies that placed a value on using information to inform people's decisions, instead of just automating

something. That's key. They tried to solve end-to-end problems. In our follow-up research, we took 30 to 35 people and interviewed them extensively in person. That's exactly what we found out. That's what it was. The companies that spent that much money were the companies that worked hard trying to get a cognitive grip on what people do for a living.

IT people don't understand what you do or what anybody in your company does. It's not their fault. They have a role and they're managed to do something. One of the things they're not managed to do is understand what you do for a living. They get in trouble with things like this because this is not an accounts payable system. This is a business intelligence system. This goes to the heart of the kind of work you do, which is far more complex than they can ever imagine.

Of the shortcomings they cite, 61 percent of them said that it just isn't relevant to the work they do. It just doesn't apply. Fifty-five percent say that the integration is poor. In other words, there are not enough data coming from multiple sources that come together that really gives them an answer. A lot of these systems are built, they're very expensive and they only pull in data from, say, SAP R/3 and maybe one other system and that's it. They don't get all of the other systems that relate to the things that people do. They don't understand it. That goes right to the heart of the silly notion of metadata. Do you know what metadata is in a data warehouse? It's just another set of relational tables. It's a glorified data dictionary. It doesn't tell people anything. It tells them where the data came from and which programmer wrote the program and what time it was last updated. Who cares about that? When you're trying to solve a problem that involves valuation or asset-liability matching, you don't care what system it came from. Ease of use and performance? Fantastic—the least critical area. That's probably because those of us in data warehousing have really knocked our heads against the wall for the last 15 years trying to get ease of use and performance to not be a problem.

But we've *really* missed all of this. Now, why did I say there's good news in here? There's a pony in here somewhere. Where is it? The pony is that the data warehousing industry is composed of practitioners and vendors. The practitioners are like all kinds of practitioners; they don't want to change anything because it's a fat cow and it's doing great. But the vendors are always jockeying with each other to do something new, to get a jump on the market, to beat some competitors, right? The people in the market-leading position don't want to change anything either. They're happy. But all these little vendors and all these brilliant people who work are constantly coming up with better ways of doing things. That's precisely what's happening with data warehousing. There's this whole notion of taking a universe of data, squeezing it, cleaning it and dumping it into a big bag called a data warehouse, where it sits forever to be used for who-knows-what purpose. It's like the best thing in the world that could happen here is that all your data have a single definition, so just have at it. There are a lot of problems with that.

Now you're seeing pieces of software evolve. One class of software is called

enterprise information integration (EII). Many years ago IBI, who used to sell FOCUS and still does sell FOCUS, came up with something it called the "virtual data warehouse." They said, "Look, we connect everything in the universe. What do you need a data warehouse for? We can just get data on the fly." It was a great idea. The only problem was that there wasn't enough time in the day to get it. You ran a query, and it might take you six years to get the data. Then you've got a mainframe system over there that is running at 99.9 percent capacity, and some user runs a query against it that just turns the lights out. That didn't work. Well, everything is different now. None of those systems are running at capacity, and adding capacity has become a relatively simple thing. The price of CPU, memory and storage has just dropped through the floor.

There is no reason to say we don't have capacity anymore, so why do we want to take data modelers, have them spend six months trying to integrate some puny little piece of data against the data warehouse, modify that data model, then change all the semantic layers and all the tools, all the universes and business objects, and all the catalogs and Cognose, when we can just run those transformations on the fly because now we have enough computing power to do it? All we need to do is have a rich enough and expressive enough environment to express those relationships and those maps, so that some brilliant piece of software can decide, "Oh well, I want to materialize that data for the next few days because it's getting hit pretty hard, but by Thursday, nobody is querying this right now, so I'm going to drop those tables and I'm going to cash these over here, because these have been getting hit pretty hard." You don't need to know about the physical models anymore because that's what slows everything down in this process. That's why people have trouble with relevance and integration and understanding. The solution for these things is on the horizon but it requires breaking the paradigm of data warehousing.

It's not going to be easy, but it's a vendor-driven business. In fact, most technology businesses are vendor driven, and the vendors are pushing pretty hard. They're making a lot of headway and I'm very hopeful.

Let me draw your attention to the URL, www.hiredbrains.com/knowout.html. There are a lot of articles I've written on these subjects, if you want to do some further research. There are links within those articles to other things that may be of interest to you. Thank you.

FROM THE FLOOR: Neil, you talked about the one situation where you kind of locked the IT people out of the room in order to get the thing done. If you truly want to start building the data warehouse by starting at the end and focusing on business results, economic value and then working backward, my experience has been that even though I say that I don't want to talk data architecture, every time I turn around somebody is up on a white board building those big charts. Can you tell me some techniques that you do to keep people focused on the end and working backward?

MR. RADEN: Only lately, because I was one of those guys who kept drawing them back to the diagrams up on the white board for a long time. I came to this realization over time. As a practitioner and as an outsider, it's hard for me to get people in the room to behave. I can try, but as a consultant, I can only guide or inform. I can't dictate. That's why you need that vision person to keep things going back to where they need to go.

MR. JONATHAN TREND: It seems like in all this, and I realize the session is data warehousing, that you've just accepted that the kind of sources that feed into the data warehouse are whatever they are. They could be coming from anywhere in various formats and so forth. Is that just something you have to accept, or would you like to see something where the feeds into it have some kind of integrated system? Much of the work I do is dealing with all this bad data. How do you deal with it? My solution is, don't let that data get in in the first place. You can't just type in garbage.

MR. McINTYRE: We run into that all the time as well. You're right. You want to go as far upstream into the process as you can. We've got a couple of huge data warehouses that we maintain and refresh on a weekly basis. When we start running some of our data extracts off of the warehouse and we find problems with the data, we don't stop there. We'll actually go further up into the system and try to find out where it is. If it's actually a source system problem, then we'll work with the client to put in a request to get that changed. We do want to go back all the way to the operating system, but again, that's why this whole process is huge. If you're only going to try to take the shortcut to getting there, it's just not going to happen. My experience has been that the reason why these efforts fail is that nobody is willing to spend potentially a couple of years, not just implementing the data warehouse or the different tools, but on the whole system. It's getting the process in place. It's finding the data problems. Often the data problems are tied into business process problems. You have to fix those as well. My challenge to you is, don't accept bad data. Find out and know that you've got bad data, deal with it for the cycle that you're dealing with, but then immediately work to figure out how you solve for that. If you're finding them and having to deal with it, everybody else getting that data is dealing with it as well.

MR. RADEN: I agree. That used to be a big controversy in data warehousing. What do you do when you find bad data? Is it your responsibility to work with the source systems and get it corrected? Or do you do your own thing and clean it the best you can and shut up? In the old days, you used to do the latter, because IT didn't want to hear about it. But to its credit, one thing data warehousing has done is raise the consciousness about data quality and data integrity. Most people in IT now understand that bad data is everybody's problem. I would say that the answer is, if you find bad data in the system, you *do* work with the source systems to get that corrected. We're not always successful. Some of it is historical data and they just can't deal with it.