1999 Valuation Actuary Symposium
September 23–24, 1999
Los Angeles, California

Session 34PD
Model Integration for Various Corporate Purposes

Moderator: J. Bradley Murray
Panelist: Yiji S. Starr

In today's environment, actuaries use models for statutory projections, GAAP projections, budgeting, corporate planning, and product pricing. Frequently, different models are used for each one of these activities leading to the possibility of assumption and model mismatch for several similar projections.

The panel discusses how to manage and integrate current models in order to minimize model and assumption mismatch and to maximize the potential of current models for multiple purposes.

MR. J. BRADLEY MURRAY: We are going to talk about model integration. I am a consultant at Milliman & Robertson. We also have Yiji Starr, who's a consultant at Ernst and Young in the New York Office. She is going to provide a perspective on model integrations and models, in general, on more of an overview and conceptual basis. I am going to give a perspective on how to approach model integration.

MS. YIJI S. STARR: When I was first recruited for this panel, I looked at the description of this session, and it referred to how we should integrate. The description made certain assumptions that integration has its advantages, and that it would be in everyone's interest to do it. I wanted to offer some considerations you might want to go through before you actually integrate. It's very difficult to offer a group of experienced actuaries a solution that they haven't thought of. By asking questions rather than offering solutions, I hope I won't offend anybody or say anything that's not right.

I'm going to be talking about a list of considerations that you might want to think about before you actually integrate actuarial systems. I'll start talking about the types of actuarial systems we have. Before you start integrating them, you ought to think about what they are, and then get a list of integration considerations. As you go through these issues or considerations, I think it will become fairly obvious as to how or what is the best way to integrate them.

Types of actuarial systems. I like to think of them in a simplistic mindset. There are two types. There's seriatim-based and cell-based systems. Seriatim-based systems include more of the administrative systems, where you use a policy-by-policy basis, such as your valuation system, for some of the traditional products in which you might have reserve factors. You just apply them to every policy, and you come up with the reserves. There are cell-based systems in which you use a modeling approach to project information about a certain block of business. In my experience, seriatim-based systems are fairly difficult to integrate. That's because they are designed to handle a large volume of data well and fast, and they do one thing well. When you try to integrate with other systems, you sometimes lose that efficiency.

I'm going to try to concentrate on the cell-based or actuarial models. There are many actuarial models within any large corporation or within companies. I choose to list the following four, which I think are common in every company. There are the pricing models, the planning models, the reserve adequacy test, and asset/liability models. These models might be statutory only. They might be pre-tax only, or they might be after-tax, statutory, and GAAP.

What are the items projected by these models? You typically project the premiums, the investment returns, expenses (including commissions and benefits), and reserves. Expenses include commissions. Benefits include mortality, surrender and withdraw benefits. The type of reserves will depend on the nature of your model. It might be statutory, it might be tax, and it might be GAAP.

Another way to look at this is actuarial models typically project the liability cash flows. Accountants often say to me "Actuaries don't know what they're doing. They call premiums a liability item." That's more the actuarial way of thinking about it. We call all of them the liability cash flows.

Each model will project the items I just mentioned, but they'll probably put a slightly different spin on it. The pricing models are typically product-based. You will have an investment return many times as just a certain interest rate. They're not dynamic, and oftentimes it's a level interest rate. You probably use unit expenses, and you have return measures such as return on equity (ROE), percentage of premium, or both. You might also have a GAAP and statutory basis, and you tend to examine results over many cells to ensure profits are reasonable for all the cells such as every five-year issue ages, male, female, underwriting classes, and so on.

For planning types of models, you might project the same information, but the spin is a little bit different. You tend to look at company bases, a line-of-business base, or a distribution channel base. In other words, what is your management unit? You tend to look at information in aggregate. You will reflect the actual in-force business and use new sales assumptions. You'll have fewer sells than your pricing model. You might use your pricing assumptions or emerging experience, depending on how long the block has been in-force and how credible your experiences are. Many times you will use budget expenses rather than unit expenses. For planning purposes, you want to reflect what you think the overall expenses are. Budget expenses are often an accurate measure. You might still need unit expenses when you want to allocate the budget expenses over a line of business, a distribution channel, or a company. You might use investment returns for the next three or five years developed by the investment department. That may not be a level interest rate. It could be up and down, and you reflect what the investment department thinks the market is.

Asset adequacy testing models were typically driven by regulation. It focused on the adequacy of the reserves. There are prescribed scenarios, plus a few others you might have developed yourself. They are typically conservative assumptions because it's statutory and solvency oriented.

The asset/liability model may be an extension of the reserve adequacy model, in some cases. It's cash-flow based. You focus more on the interaction between assets and liabilities. Similar to the reserve adequacy model, you need to have the ability to model assets well, which is different from your pricing and planning models in many situations.

With all of this said and done, what are integration considerations? I like to look at it two ways—as a vertical integration or as a horizontal integration. A vertical integration is one where you might be trying to integrate all of the models within the same line of business or the same product group. In other words, you want to have one model for all purposes: pricing, business planning, or whatever, for one line of business. The horizontal integrations were for large companies and multi-line businesses where you might want to have one model, one purpose, business planning or wherever, that's used by all business units. Either way you want to figure out the common elements. It's fair to see liability cash flow that might be one common element. Sales you used might be another common element. It varies by how you use your model and what processes are involved.

What are the assumptions? Models use different assumptions, rightfully so. For business planning purposes, you'll probably want to use best-estimate assumptions. For reserve adequacy analysis, you might want to use conservative assumptions. How do you want to integrate those assumptions and cross-business assumptions?

Finally, let's discuss modeling methodologies. How do you want to integrate that? Do you want to project everything on a monthly basis or an annual basis? Do you want to model it in a spreadsheet? Do you want to model it in a home-grown system or a vendor system? That's probably the most difficult item.

Now let's discuss the advantages. You'll have efficiency because you'll have one big model that everyone uses. It's on the common platform, so you make sure the methodologies are consistent. You probably have to do less reconciliation work. You can communicate the results easier. Everyone knows what the model is supposed to do. The last item is a key for a great many large organizations—the ease of knowledge transfer. When you have a student rotation program,

where students go from department to department, having one, big model that they are familiar with makes it easier for them to just pick up from where they left off instead of having to learn a new system in every area they go to.

What are the disadvantages? It may be inefficient. It seems sort of like a paradox. It's efficient to integrate, and it's inefficient to integrate. While you're integrating it, you might have some inefficiency. Even after you have integrated, you might still have some inefficiency. It's because you might have lack of flexibility within your model. A model that's designed to do one thing might not do something else as well. You might have a lack of specialization. The pricing and the business planning models that we often used don't really have the asset specialties, so they might not be appropriate for the asset/liability models.

Finally, everyone probably has lots of value in their experiences. You've reconciled different model results, and you could be spending a great deal of time coming up with a conclusion that both numbers you are presented with are correct, but they're just different.

Let's talk about how I'm going to present my views on how you want to integrate models. There is an easier way to do it or one common sense approach to figure out the common denominators. If you are doing a vertical integration, where you have one model for all purposes for your business unit, you might want to build enough basic common grounds and common elements, and then allow different people to use it differently. In other words, build a base model, and then expand it for different uses.

If you were doing a horizontal integration, you might want to define common desired outputs from each of your business units, over which you'll consolidate at the corporate level. Some of these common required outputs are liability cash flows and reserves, tax, GAAP, or statutory. You might have other needed information. One thing that comes to mind is the deferred acquisition expense. In order to do GAAP analysis, you have to split your expenses into deferred and nondeferred. You also need enough information to calculate your estimated gross margins and estimated gross profits.

Assumptions are probably the easiest and most difficult item. Every model uses different type of assumptions, whether they are statutory or GAAP, conservative or best estimate. Brad will talk a little bit more about that later. All I want to say is it needs a lot of coordination, whether it be through technology, even through something low tech. Everybody should talk to each other and make sure they understand the appropriate assumption for different purposes.

I think I alluded to something many items in the modeling platform. It's very difficult to choose one that does everything well. There are many vendor systems out there. I think all of them have some strengths and some weaknesses. Whichever platform you choose, you must have enough flexibility to allow you to accomplish all your modeling needs. Finally, one of my favorite ways of doing it is what I call the back-end integration.

This is a type of horizontal integration. We have done this for a couple of clients. We have used it most successfully for business planning functions, where each business unit produced our business-unit-related results. They used various different software or systems or sometimes spreadsheets to project the information for their business unit. The health line business might be projecting results using a completely different system from the life side. Within the life side, the way you would project traditional is different from term and universal life. You consolidate all of the outputs from these business units at the end or at the back end.

At that point, you can incorporate your cross-business assumption, such as investment income, tax, and budget expenses. If the business unit provided you with information at the lowest level of detail, such as product, distribution channel, company level, or combinations, you can, at the consolidation stage, project different cuts of it. For example, you can produce reports by distribution channel, by a statutory entity, and by line of business, or by anything else that's useful for management to use in analyzing its results.

I went through a lot of this very quickly, and I probably didn't give you any how-tos. I just wanted to go through this thought process to raise a lot of issues and to encourage you to think about how you might want to integrate your models. I will turn the discussion over to Brad who will tell you how to do some of this stuff.

MR. MURRAY: I think Yiji has given us a good appreciation of some of the concepts that you would want to consider when looking to integrate models and how you might go about that from a conceptual level. My aim is to take that perspective and talk a little bit about how we would actually go about integrating models. One of the areas that I'm going to have a strong emphasis on in my discussion is technology. The Internet is such a big thing. I hope to give a somewhat interesting angle on this issue.

I'm going to start by presenting the overview of what I'm going to go through. We're going to look at what I see as the barriers to model integration, which does not fall into some of the solutions that exist out there, especially where we are with technology today. Those would be things such as data integration and data warehousing.

We all know a major issue with models is data. I know from my experience in working with clients how amazing it is that resources are allocated to dealing with data issues. I hope to give you some ideas here that will help to solve those issues and to get at what we can do on model integration. I'm going to get into a conceptual construct of unbundling and how to view your model structure. Then, look into another areas of model simplification that I think many of us can appreciate. As I said before, I'll be emphasizing technology throughout my discussion.

As Yiji said, if you are going to get into model integration, there's a need for technology. The need might be for something high tech, where you use technology to do the communications and coordination, or low tech, where you just need people to talk to each other. I think documentation is a good example. You always hear about how important documentation is, and we really see that value when you're picking up that model a year ago or looking at something you did. You're scratching your head and somebody's asking questions. We know the value, but how often in any of your organizations do people have robust documentation processes? I argue that if you build the field, they'll come to you. If you have the means and tools to make this kind of stuff easier, it will happen. E-mail is a good example of that.

Let's get into the barriers that exist to model integration; first, there are large complex models. We have these huge models that evolve very quickly with these nifty actual projection systems.

These have been made available to our profession in the last decade. There is fragmented source data and modeling inputs. Where do sources come from? Assumptions come up. Where is an in-force file coming from? I went to the IT department, and I gave them my data requests. I guess that what they gave me is right. I'm exaggerating to make a point, but I think we can appreciate what I'm talking about.

Poor communication goes back to the coordination. It applies even on single models or models specific for a function or a business unit. There is difficulty in sharing data. Let's really try to share data across business units. The means to do that are not really always in front of you. You're required to save files, to send people e-mails, and it just becomes too cumbersome to really do that.

I say technology is also somewhat of a barrier, and I'm a big fan of technology. I actually think that technology has been part of the problem. Consider the modeling process and start from the back end. I'm going to use the term back end and front end more from a functional perspective. Yiji referred to it from a business angle.

The back end is where all the data are consolidated, and the front end is where all the modeling and reporting occurs. Systems like Alpha, TAS, and PPT are very high tech. They have all the bells and whistles. We get in these systems and just want to use all of the features available to us.

I would argue this is what leads us to building these huge models and these complex models and that people can get their arms around. The focus you're cramming and just trying to get everything into the model. You have that five minutes before you're looking at results saying, "Yeah, that number looks right. Just take the decimal out three more places and we'll be set." I was talking to some folks recently about doing cash-flow testing and some things. They described the process as just following steps to build the models and never having a chance to understand what they're doing, so I think that has hurt us.

There is another side of it. Data issues and data manipulation is considered low tech. I'm going to talk about some of the tools as a contrast. We use low tech-tools to manipulate data. Data are often transferred via text files yet higher tech tools exist, such as using databases.

How might we eliminate these barriers and make some progress on this? In terms of model integration, the key is data integration, at the bottom of the process; then move your way upwards to building your assumptions, getting at your model points, building your model, and then reporting results. There is a common denominator across the board. I think Yigi brought up a good point when looking at integrating models. You don't necessarily want to integrate models in people. That's not a desirable result, and you do want to think about things before you go about doing them. Data are common. You must start from the same point. You'd like to think that everybody is from the same pool of data. If you integrate in this area, I think you'll realize these great benefits. Another two areas are model structure unbundling, and that's basically just taking the model and pulling it apart so that it's easier to work with. The other area involves simplifying the modeling process.

Data integration. What does that mean? There's consolidation and validation. The key word is validation. It's doesn't always exist until after the fact. The results are out. You need to present your results outside of your area, such as another "business unit or a different function." A typical challenge is to prove to these outside people that your results are right. You might look at some validation measures and find you are not validating. You quickly arrange your model points to make it validated. This validation step occurs to bite in the process. Validation occurs in the beginning of the modeling procession.

Let's go back to communication and sharing of data. You need to be able to have people access this information. It's data warehouse technology that I think is very exciting, and it has a lot of promise in this arena.

What do we mean by a data warehouse? The data warehouse is basically a repository for data. It's a simple concept. It's really where you're able to take your data from all your various sources and consolidate them up front. This is different from having to request similar data for

different needs such as experience studies, model building, or financial analysis. In the data warehouse approach, you first identify the needs and then collect the data. Various systems would feed into a corporate-wide data warehouse.

This data warehouse technology is something that many insurance companies are beginning to embrace. Insurance companies are very intensive with data and haven't really done a very good job of leveraging off of that. As an extension to this corporate data warehouse, you can have what's called a data mart, which is similar to a grocery store. You have your wholesale stuff at the back end, and then you have the data mart that would contain the items that actuaries would be interested in.

You get comfortable with the data at this level, and not at the modeling point. You know what's good, and you know what's bad.

One of the great characteristics of data warehousing is the common naming convention. We all know how many times we've been down that road where you go to systems and say, "I'd just like the premium amount for the next year." The next thing you know, you get a stream of numbers and somebody comes and shows you their "next year" premium numbers. Both sets of numbers differ like night and day.

How can we better understand what drives a data warehouse? One of the things that might come to mind is what a huge effort it could be. I'm going to give you a perspective on what drives a data warehouse. I think the technology and the concepts are something that can be scaled down for an area or a small unit to use. You do not have to wait for the whole corporation to move over to it.

We know a database is a two-dimensional table. You have your fields as your columns, fields of information, and your row or records. A relational debase is really a collection of databases. It's multi-dimensional. It comes from a two-dimensional and a multi-dimensional framework.

To give a better picture of what a relational database looks like and how effective it is, I have an example of what somebody might do if they're doing experience studies. You have various sources of data. We have administration systems, claim systems, and areas from which you are going to need to pull off different pieces of information. There are common fields to each of those various systems, and there are data sources. We have a policy number, and you can see where a policy number is common to the various areas. In using those common fields, you're able to have the data remain in its home base. You're not required to pull it all together.

One of the things that we do in model building is we just define a huge extract and pull everything together on one, large, flat file. That's not a very efficient way of storing the data. The data warehouse approach allows one to pull what is needed from the various sources.

What are some of the advantages of what this whole process does for you? First, you're able to leverage off your current data sources. You don't need to go about making a whole bunch of different extracts in all these various side summaries of your data. You're able to keep the data in their root form and use them very effectively. The structure is very conducive to universal access. It gives you a mechanism for people to access the data. I'm assuming most of you have dealt with Access or Fox Pro, so you might understand how you would query information to be able to access the data and slice and dice it. You're able to drive multiple processes from the same data source (I'm referring to the repository). You've consolidated your data, and it's in one spot. You know what's good, and you know what's bad. You understand your common terms that you've identified for the fields. You can drive your modeling from that process, even if you want to have all your models reside in their various areas or functions and even for analyses and duration of your assumptions. Doing this is pretty powerful, even if you have a GAAP model and a cash-flow model or assumptions. At the very least, you started from the same spot and you have common terms that you're going to use in terms of describing those data. I think that really makes the reconciliation processes (in terms of understanding the various models and processes in the actuarial area) so much easier.

Some of the other areas deal with the data functions on the back end. You do it once. You don't keep going back and forth to the IT department to get your information. You have your data

store. It's easy to integrate multiple data sources. By working with IT, you are, in effect, asking them to give a snapshot with a subset of one system or another. You don't need to go through and ask them for specific information on your needs. You look at the raw data and how that raw data would filter up to what you need to do. You have all your characteristics and are able to capture lots of information in this format. You could start small and easily expand. If you think about it, you can even start off with a subset of an administration system and valuation system. Over time, as you might have more needs that come up, you can just piggyback further data sources into that repository and not have to go backwards.

One of the things I wanted to point out here is what I call a quasi-data warehouse. It is something we can start right away. We do not have to wait for this large mandate to take place across the enterprise. As I said, a corporate data warehouse is a simplistically relational database structure. Don't mention relational databases and data warehouses in the same sentence to a die hard data warehouse person. A data warehouse is a much larger step up compared with relational databases.

Nonetheless, imagine taking off your desktop Access and Fox Pro and using the relational database concepts that we talked about before. It wouldn't take much for your local area network (LAN) to kind of allocate an area to store data of this nature and to be able to start dealing with your IT and your various data sources in a much more efficient manner. Grab the data first, pull it in, and store it in.

This approach would get you all the great benefits of data warehousing, but you would benefit from central location and universal accessibility. It would be a great facilitator to share information, so it would go very far in meeting a lot of the issues that we're talking about here for model integration. To move from this relational database approach to a full sledge data warehouse, you start getting into a multidimensional type of structure and various sophisticated front-end tools that allow you to drill down and do pretty high-end analyses on those data.

What I just described to you would be something where you might do a simple PC or a drive on a LAN. A true data warehouse would at least require a dedicated LAN, with a SQL server or database server that would reside on that LAN.

I'm going to go into the other areas that I alluded to earlier in terms of how we can improve model integration. Instead of modeling every five policies with a cell, you might want to start small or start from a macro perspective. Choose a couple of major plans that make up whatever business that your modeling would have and map everything to those few plans. Go through all the steps you would typically go through in model building, except don't spend so much time on cramming information into the model. Instead, spend more time getting your arms around validating the model itself. You can sit there and quickly get to a validation point doing dynamic and static validations. You're not going to achieve 1% or 2% thresholds. But you should be in the ballpark of 10% to 20%. This, at least, makes you comfortable that the model was going in the right direction.

Once you have this core model, it can serve as kind of an interchangeable piece for other areas or for other needs. A full-blown model would just require additional detail to be layered on top of that. If you ever have to go back or if you have a problem in understanding what the results are, you can strike that additional information and go back to the core model. Now that you're calibrated, it's about doing what you think you should be doing.

The next issue and next solution is unbundling. I think this is something that really is more of a constructor or a way of viewing your model structure. Consider this in two aspects. One is kind of the functional part of the model. A model starts off with cash flows and on top of those cash-flow projections, we layer on accounting convention, such as GAAP, statutory or whatever. Once we have our accounting convention there, we then layer on some reporting process in terms of how we want to view that information. This sounds kind of simplistic, but thinking about it in these terms might help you to better understand how you might go about pulling a model apart and keeping pieces that are common across business units or common across functions. Then replace parts of it that you think are specific to the need at hand.

The other area of unbundling on the model input side is the assumptions. All your assumptions and calculation routines can be thought of as modules. Benefits of unbundling include adaptability. If you're able to pull pieces out of that model, interchange it with what you think is required for the specific issue that you're looking into. It's easy to keep pace with technological improvements. We have models in which we kind of get everything in one big bundle. We look at the reporting function, the projection system, and the assumptions. Everything is kind of bundled together. You pick a vendor to do that in terms of the various projection systems. If you look at it from that layered perspective, cash flows, the accounting system, and reporting, you're able to have specific technology for each of those areas. We can appreciate reporting. You can report easily from a spreadsheet approach (which sometimes is very desirable) as opposed to thinking that you're all stuck with the same approach across the board.

It's half of these various layers. This is kind of an example of splitting it apart. You have the specific engine driving your various areas and the layers represented. The bottom layer represents your cash flows. The middle layer represents your accounting, convention, or system. The top layer is your reporting. It can give you a flavor of what I'm talking about in terms of plug and play and removing areas where you think Excel is working great for you to report. Maybe a graphical type of interface will be better tomorrow. You might find that there are new developments. Due to technology, there are developments happening at a rapid pace. You can always keep pace with that as opposed to taking your whole process and saying, "Should we replace our whole projection system to be able to keep pace?" That's an issue with which companies are always grappling.

The other part of this (what I call the model form) is you have a way of structuring the model for your various purposes. You can look at the accounting layer to say I want to do GAAP or embedded value. How do I want to look at this, and how do I want to account for our financials? On the capital side, we would do assets and liabilities, and maybe a liability model or whatever capital that you're aiming to focus on.

When reporting your numbers, you could use income statements, balance sheets, and things of that nature. I hope this encapsulates everything that I've spoken about. On the back end, we

integrate the data through a data warehouse approach. We integrate the modeling process through unbundling. This can be thought of as a puzzle. We break the modeling process into pieces such that it's easier to use the common components for multiple modeling tasks.

The strengths on this data warehouse is you can take your pieces and, in that sense, your lapse studies, or your mortality assumptions, or your model points, and actually keep them stored in the data warehouse and provide something. Let's say the pricing area builds a model and has some basic sells that they started out with. Down the road, you want to build an in-force model to look at management business. You have the data stored there. You can go back, and you do not have to start from the ground zero, again. We need to do this. You have to start from scratch. It's a bit exaggerated. I hope it's not that bad all the time. I think we can appreciate something like that.

I hope this has given you a different angle in terms of looking at this problem and using the technology and looking at data warehousing. If those kinds of initiatives are taking place in your company, try to get in on those conversations. I found that many of the data warehouse vendors don't want to come near the actuaries because they're completely intimidated by that area. So if that is happening at your company, becoming part of that process would benefit you greatly.