

Modeler Q&A With Lei Rao-Knight

By Ben Neff

Lei Rao-Knight, FSA, MAAA, is vice president and actuary in the Modeling Center of Excellence at Prudential Financial. She graduated from the University of Connecticut with a Master of Science in Actuarial Science. Lei has 20 years of actuarial experience spanning pension consulting, valuation reporting and modeling for individual life, annuities and retirement.

Q: What kind of modeling work do you do, and what software platform(s) do you use?

We focus on asset-liability management modeling for the individual life business using Moody's AXIS. The main responsibilities include:

- Gathering or reviewing business requirements
- Model design and development
- Model testing
- Production model implementation
- End-to-end process development and run automation

The models are used for pricing activities, statutory reserve reporting, forecast projections, asset adequacy testing, replication portfolio cash flow projection, capital management and other internal management purposes.

Our models are often used for reporting purposes, and these need fast turnaround times and robust controls during production.

Q: Since you use software that is vendor-maintained, can you describe what you mean by model design and development and what, if any, model coding you need to do?

We cover the following model design and development:

- Defining input requirements including input data, rate tables, assumptions and other key switches that are used by the AXIS model
- Creating tools if needed to transform required inputs to the format that the AXIS model can utilize
- Translating business requirements into technical specifications in AXIS
- Evaluating AXIS capability and redesigning the modeling methodology to fit in AXIS existing structures or to request modifications
- Determining model configuration in AXIS such as the structure and contents of the model including funds, sub-funds and cells
- Creating new tables, including coding in formula tables
- Designing batch process to improve the model runtime or run automation
- Designing output solutions to meet business requirements

Q: How do you plan or prepare for a modeling change?

The request for a modeling change typically goes through an intake or prioritization process. After key stakeholders reach an agreement to proceed with a modeling change, we follow a well-defined model development life cycle (MDLC). The complete MDLC generally includes the following steps:

1. Prepare business requirements for the model change
2. Model design based on business requirements
3. Model development
4. Model testing
5. Production model implementation

Our modeling team is responsible for steps 2 to 5, while users are responsible for business requirements in step 1, user acceptance testing in step 4 and production support in step 5.

Q: When you find a bug that has an immaterial impact on results, can you let it go or not?

The answer is case specific, and it can go either way. These are general steps we take:

1. If the bug has an immaterial impact on results, we typically reach out to key stakeholders who use these model results and jointly reach an agreement on whether it needs to be fixed.



2. If the decision is to fix it, even if it causes an immaterial impact, we will determine when will be the proper time to change the model for this bug.
3. If there is ample time to incorporate this fix in the future model development cycle, we will treat it the same as one of many model change requests and follow the MDLC in the future model development cycle.
4. If it is time constrained, we will follow a simplified version of the MDLC by skipping unnecessary steps such as preparing business requirements to speed up the development cycle.

Q: Documentation—as you go along or after you’re done?

When we follow a well-defined MDLC, documentation is defined and created during these key steps:

1. Business requirement documentation is produced when users prepare business requirements for the model change.
2. Functional specs are generated by the modeling team when they work on model design based on business requirements.
3. Technical specs will be finalized when the modeling team completes related model developments.
4. Testing plans and testing files are created when both the modeling and user teams perform various model testing.

5. The user manual is updated when the new version of the production model is implemented and released. In addition, sign-off and model affirmation will be also produced at the end.

Q: Do you have a modeling pet peeve?

My pet peeve would be insufficient documentation of coding. This really makes it difficult for others to review or understand what the code is trying to achieve.

Q: What’s the most frequent piece of code or software feature that you can never remember the syntax/setup for?

The most frequent piece of code that I can never remember the syntax for is coding to connect external files within formula tables in AXIS.

Q: What was the last problem you encountered that had an easier-than-expected solution?

One example that my team encountered recently was we needed to build a blending rate table in AXIS. However, it appeared to only allow equity index selections but not bond selections. It turned out that if a link to bond is created, this link will then be available. Without this workaround, we will need to use formula tables or explore a different modeling approach. Lesson learned is that AXIS is quite flexible and it allows for creative solutions.

Q: What's something new you picked up recently that you'd like to share?

Our models are often used for reporting purposes, and these need fast turnaround times and robust controls during production. Having a well-thought-out vision of the complete end-to-end process upfront is very helpful for large conversion projects. It can provide good guiding principles for the model design and reduce redesigns of models later to meet requirements of the end-to-end process.

Q: Can you share a favorite time-saving coding technique or feature of your current software that you like to use?

One member of my team shared that R is a lot faster than Excel or VBA [Visual Basic for Applications] at generating random numbers (for sampling, simulations, etc.).

Q: Have you seen any dangerous or inefficient coding or model configuration practices that you would advise other modelers to avoid?

Our team raised a few examples, which include hardcoded plan codes and assumption tables, and setting a variable and overriding it later within a different set of code.

Q: What is the funniest or most surprising modeling error you have encountered?

These are a few examples shared by the team:

- Copying a blank field to SQL and SQL changed it to NULL.
- Lookup functions in AXIS let you specify a default value when the lookup fails. In our initial model build, the developer used the string “.NULL.” as the default value, instead of just assigning the variable the value .Null., which is how AXIS denotes null values. If you specify a null value when assigning tables to cells, it will skip that record and not try to assign anything, but when you code the string “.NULL.” the system will treat that as an actual

table name and look for a table called “.NULL.” This leads to some interesting model errors. This was frustrating to debug because just from looking at the table, it was hard to tell if AXIS was using a null value or a string called “.NULL.”

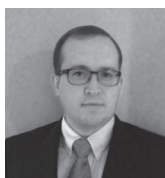
- Language used in AXIS queries is different from VBA, where zero means false.

Q: What do you wish consultants understood about your models?

We have good relationships with our consultants and they often come up with quick solutions to solve model issues. I look to consultants to balance the need for a quick solution with long-term considerations in the following areas:

- Does it require any manual updates to maintain the model for the future? Ideally, we would like to avoid manual updates for the routine production updates.
- How easy is it to expand to other products or usages? It will save overall time if a solution doesn't require major redesigns every time the model is expanded.
- How does it fit to the requirements for the end-to-end process? We are driving a more controlled and automated end-to-end solution especially for many production reporting purposes. Some solutions work well for ad hoc situations, but not for the production model.

Overall, it will be more effective for the consultant to understand our long-term goals so that we can work together to develop a more robust model. ■



Ben Neff, FSA, is director and client relationship actuary in the Enterprise Risk Solutions division of Moody's Analytics. He can be reached at Ben.Neff@moody.com.