



Article from

**The Modeling Platform**

November 2017

Issue 6

# Actuarial Model Component Design

By William Cember and Jeffrey Yoon

As managers of risk, most actuaries are tasked with answering questions about how things will play out in the future:

- How much money do I need to set aside to meet the obligations to my policyholders?
- What should the charges be on a new product to make it profitable yet competitive?
- What will the capital position of my company look like 10 years from now?

In answering these questions, a primary concern is the integrity of the calculations and data used in our analysis. With faulty calculations and poor data, we cannot give reliable guidance to our stakeholders.

Just as important as the *what* in what we do is the *how*. For actuaries, the *how* is our models, and just as we need to make sure those models are programmed correctly to calculate the metric we are interested in now, we also need to make sure they are well designed so they will continue to be reliable in the future.

In this article, we define and discuss the components of actuarial models. We pose key design questions as well as the criteria used to answer them. We also provide you with tools to not only build the *what* but also design the *how*. This will help ensure that even though data is updated and questions change, actuaries are still able to obtain the correct calculation.

## ACTUARIAL MODEL COMPONENTS

The three components of an actuarial model—input repository, calculation engine, and output repository—have to work together in harmony. Each has an important role to play in finding answers for our clients.

### Calculation Engine

The first thing that comes to mind when most people think of a model is the calculation engine. This component performs the core calculations, turning input data into management metrics. Some example functions performed by actuarial calculations include the following:

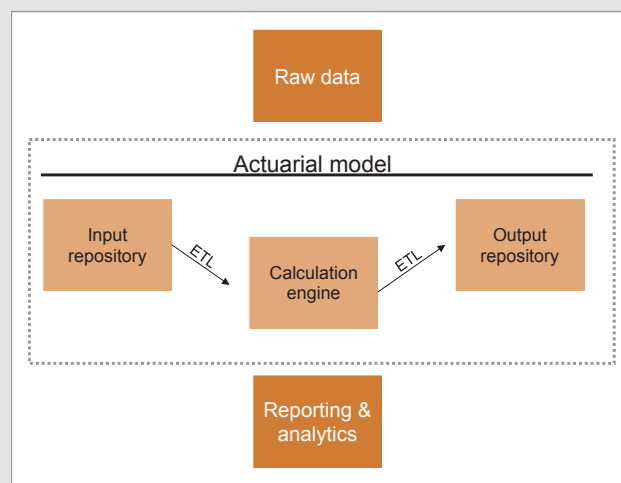
## WHAT IS A MODEL?

If you ask 10 actuaries what a model is, you most likely will get 10 different answers. The Federal Reserve’s supervisory guidance on model risk management for large financial institutions defines a model this way:

[T]he term model refers to a quantitative method, system, or approach that applies statistical, economic, financial, or mathematical theories, techniques, and assumptions to process input data into quantitative estimates. A model consists of three components: an information input component, which delivers assumptions and data to the model; a processing component, which transforms inputs into estimates; and a reporting component, which translates the estimates into useful business information.<sup>1</sup>

Per this definition, a model does not only include the calculation engine, which is what is often thought of as the model, but also the end-to-end process (see Figure 1). This includes an input repository, output repository and the associated extract-transform-load (ETL), which passes the data back and forth.

Figure 1  
Actuarial Model Components



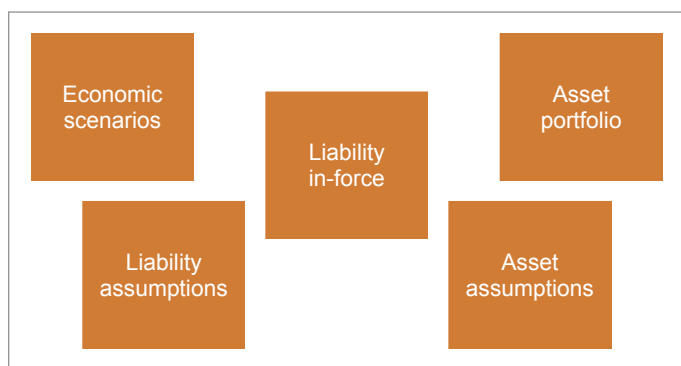
- Calculating reserves
- Projecting premiums and claims
- Projecting out assets against liabilities
- Determining required capital

Within the calculation engine, the model developer programs the methodology used to determine and project balances.

## Input and Output Repositories

The input repository stores the inputs for the broader actuarial model (see Figure 2). Depending on the maturity of the model and type of input, the input repository does not necessarily need to be in a separate location. For example, while many models will have a standardized location for the in-force inventory, assumptions are more often hardcoded in the calculation engine, and therefore, the calculation engine and input repository may be one and the same (although we don't recommend this).

Figure 2  
Input Repository



The output repository stores model output before it is used for reporting or analysis. Like the input repository, depending on the type of model output, the output repository and calculation engine may be one and the same. Decisions should be made regarding development of a separate output repository.

As models mature, first-class input and output repositories can serve purposes beyond simply storing data:

- **Approval tracking capabilities.** Before inputs can be used in a model, they must be reviewed and approved. This effort can often be manual but may be prone to error. A first-class input repository includes built-in approval tracking, ensuring the right inputs and automating the process of producing the corresponding documentation.
- **Platform independence.** The input and output repositories can be built so they are independent of the calculation engine. This allows first-class tools to be used as a backend for these repositories, reducing the need for model conversions from one actuarial platform to another.
- **ETL automation.** Automating the data ETL processes between calculation engine and input/output repository increases the ability for actuaries to focus on providing analysis and business insight, rather than performing data work. Results can then be delivered to customers faster, allowing real-time decision making.

- **Metadata.** Beyond approval tracking, the input and output repository can be designed to store rich metadata regarding when and how stored data has changed and who made the changes. This also has second-order consequences, allowing the owner of the repository to quantitatively answer questions such as whether the data are being delivered on time or how long models are taking to run.

## KEY DESIGN CRITERIA

Using the following key design criteria, a company, based on its specific characteristics and requirements, can make decisions around the design of its actuarial models.

### Accuracy

As actuaries, we always strive for the most accurate models possible. With everything else equal, we believe the more accurate a model, the better. In practice, one often must make a trade-off between accuracy and other characteristics:

- If a certain calculation is improved but the model takes three times as long to run, is it worth it?
- What is the balance between accuracy and maintainability? If it is necessary to code a model in a messy way—which is likely to break down the line—to get a calculation perfect, is it really worth it?
- Can a general solution for many similar products suffice if it is less accurate than a separate solution for each product?
- If the current approximation is replaced by a complicated solution, is the impact material enough and worth the effort?

The answers to these questions depend on the function of the model. For example, valuation models are going to have a lower threshold for errors than pricing or forecasting models. Generally, there is a trade-off between short-term and long-term accuracy. As much as we want to perfect calculations, more complicated models limit us in terms of improvement and increase the chance of future inadvertent model errors.

### Controls

Controls regarding models pertain to the process around which changes can be made to the model and how models are run in production. The actuarial model must be controlled to the extent required by the intended model's purpose. Models used for financial reporting or reserving are often subject to specific regulatory requirements, and even models used for other purposes should be subject to a defining set of controls.

At the same time, model control must be balanced against flexibility. In particular, models that are used for multiple purposes will often have users requiring differing levels of control and flexibility.

## Flexibility

Flexibility is the degree to which model users can easily achieve the business goal with their model. For example, how difficult is it to run the model with an alternative in-force policy or alternative assumptions? How easily can changes be made to the model in the future?

Different functional purposes have different requirements for flexibility. A model used to project in-force business will often have product features well defined, while a pricing model will often require the ability for the model user to implement innovative product features as these products are being designed.

## Testability

Testability is the degree to which models can be tested. For example, does a model show the underlying calculations for each step in a reserve calculation or only the final number? Similarly, how granular are model results—just what’s needed for reporting or granular enough to allow the model user to drill down when something goes wrong?

In a perfect world, our models would show every step in every calculation (full transparency) and allow the model user to drill down from aggregate to policy-level results. In practice, as we make a model more testable, it becomes less efficient. One possible practice is to build models in a flexible enough fashion so that model users can make this decision when they run the model or allow different model functions to have different degrees of testability.

## Efficiency

Efficiency is the degree to which a model can quickly perform the calculations it needs to perform using the minimum resources (computer and human). When thinking about efficiency, consider the end-to-end process of receiving final reports from model inputs rather than just how long it takes the calculation engine to complete a run. With all else equal, we want our models to be as efficient as possible. In practice, however, there is often a trade-off between efficiency and other characteristics, such as accuracy, maintainability and transparency.

One question to ask when making this trade-off is whether the extra efficiency is useful or not. For example, does it really matter whether a model takes 10 hours or 11 hours to run? Probably not. In both cases, the model runs overnight, and results are ready for the actuary to review the next morning. At the same time, it does matter whether a model takes 10 hours or 100 hours to run.

## Transparency

Transparency refers to the degree to which the underlying calculations of the model are viewable by the model user. It is always better to have a more transparent model to allow the



actuary to drill down into calculations as needed, such as when validating the model or trying to understand why the model is producing the results it is producing. In reality, though, there is often a tension between transparency and some of the other characteristics listed here, such as control and efficiency. When designing or building models, one often needs to evaluate how important transparency is for a specific business purpose against other characteristics.

## User-Friendliness

User-friendliness is the amount of training and documentation required for a new model user to run or view the model or for a new developer to make changes to the model. We want to minimize the amount of training required to interact with the model. Even if the model is producing correct results, if no one except one “expert” in the company can understand it, is it really serving its purpose?

## Standardization

Standardization is the degree to which conceptually similar pieces of the model (or the set of models within a company) are designed in similar ways (such as following a documented convention). Standardization makes models more maintainable and repeatable. It also allows model users and developers to more easily and quickly understand what the model is doing and to change the model if necessary. As obvious as it seems to standardize models, this requires up-front work to determine the model standards and discipline to enforce them down the line. No one single standard will be perfect in 100 percent of the cases.

## KEY MODELING DECISIONS

Before any model can be created, decisions need to be made that dictate what kind of model is desired and which characteristics will be of most importance to the practice at hand. The following list outlines the types of decisions that need to be made and what factors to weigh in those decisions.

- **Coupling.** Coupling refers to the degree to which components are dependent on each other. As an example, inputs that are stored directly in the calculation engine are tightly coupled. Tightly coupled architectures are often easier to build but can be harder to maintain and limit the flexibility of the model over time. For example, it is much easier to test the business impact of new assumptions if this can be done by swapping in a new assumption input file for an old one and rerunning, rather than having to go into the calculation engine and manually change internal model tables.
- **Data transformation.** As actuaries, our first instinct is to do everything ourselves, and that often involves using familiar actuarial software. Data transformation done within the calculation engine is often easier to build out by the actuary, providing more flexibility. At the same time, this creates a more tightly coupled architecture, and without careful planning, it can easily lead to a tangled nest of fragile, intertwined data manipulations mixed with calculations. Separating out the data transformation from the calculation engine allows us to use best-in-class tools that are specifically optimized for manipulating data.
- **Modularity.** For a single line of business, should there be a model that projects out both statutory and GAAP reserves or should these be separate models? In the abstract, it's easy to say that our models should be as flexible as possible and we should always be building out the more general solution, but that can be difficult from an engineering perspective. Often there are nuances to various calculations, making it difficult to build a “one size fits all” solution. Similarly, from a process perspective, it's very often the case that separate teams are responsible for various calculations, and building a model that does both calculations requires harmonizing the modeling approach among teams. Often, more modular models are more expandable for new products or methodologies, as already built components can be leveraged.
- **Open vs. closed systems.** Actuarial platforms that are locked down—or are closed systems—allow for the use of a vendor-created solution that has been validated and tested. Conversely, open systems allow a company's actuaries or dedicated developers to code more parts of the calculation

to meet company-specific requirements or enrich diagnostic elements; this approach offers more flexibility and transparency at the risk of being less controlled.

- **Reporting vs. analytics.** Ideally, our models would produce perfectly granular output that would allow us to drill down and across the data in every dimension possible on demand. Our models would be forever error-free and run instantaneously. In reality, there is an inherent trade-off between calculation efficiency and output granularity, and we need to be able to strike a happy balance.
- **Enterprise-level standardization.** Just as an individual model can be standardized, the modeling function across a company can be standardized. This can range from harmonizing the actuarial software and model design standards across the company to organizational design of the modeling function. The more standardized solutions will allow “plug and play” and provide compatibility with other systems, promoting a fully automated end-to-end process.

In the preceding pages we defined the actuarial model and discussed the characteristics of a good actuarial model. When we know what makes a good actuarial model, a common set of criteria can be used to make the decisions required to design, build and maintain these models.

We also presented a few of these decisions, but we did not provide the “right answer.” The right answer depends on a company's individual needs or a department's specific requirements. The criteria can assist in selecting the right approach for a company or department. ■



William Cember, FSA, MAAA, is a director at Prudential. He can be reached at [william.cember@prudential.com](mailto:william.cember@prudential.com).



Jeffrey Yoon, FSA, MAAA, is a vice president at Prudential. He can be reached at [jeffrey.yoon@prudential.com](mailto:jeffrey.yoon@prudential.com).

### ENDNOTE

- 1 Attachment to SR Letter 11-7, “Supervisory Guidance on Model Risk Management,” Board of Governors of the Federal Reserve System, April 4, 2011, <https://www.federalreserve.gov/supervisionreg/srletters/sr1107a1.pdf>.