REPRINT

R&C risk & compliance

DRIVING RISK MANAGEMENT WITH A ROBUST MODEL IMPLEMENTATION PLATFORM

REPRINTED FROM:
RISK & COMPLIANCE MAGAZINE
OCT-DEC 2017 ISSUE

www.riskandcompliancemagazine.com

Visit the website to request
a free copy of the full e-magazine

Published by Financier Worldwide Ltd
riskandcompliance@financierworldwide.com

PERSPECTIVES

DRIVING RISK MANAGEMENT WITH A ROBUST MODEL IMPLEMENTATION PLATFORM

BY **ERIK LEAVER AND TRAVIS GARNER**
> SAS INSTITUTE, INC.

Predictive models have long been an integral part of effective risk management in financial institutions. However, the 2008 financial crisis brought the usage of these models to the forefront of regulatory scrutiny. In the years that have followed, many institutions have endeavoured to improve both the quality of the models that they employ as well as framework within which they implement their models. Moreover, new regulations like IFRS9 and CECL will require more complex and granular modelling frameworks. Not only does this require increased statistical rigour, but much greater transparency around the entire modelling process has become a necessity. Financial institutions of all sizes face many challenges related to ramping up the quality of their modelling frameworks.

Risk models are highly scrutinised by many different groups, such as internal and external model validation teams, internal and external auditors, regulators, senior management and business users. Since this level of scrutiny necessitates many costly reviews, reducing frictions in any of the review processes can lead to considerable cost savings. While the first wave of scrutiny focused more on model estimation, increasingly focus is shifting to the way models are

implemented –particularly with an interconnected system of models.

Many large firms have historically made a delineation between model estimation and model implementation, with completely separate teams handling each. The reason was that model developers were typically stronger in fields like econometrics and statistics, while the model implementers were stronger in fields like computer science, programming and financial engineering. However, this setup can lead to problems in hand-offs between these two groups, such as incorrect translations between the two teams, long timelines to get models into production, inability to make changes or improvements to models, and constraints on building best-in-class models due to knowledge gaps in either group.

Conversely, both large and small firms face challenges when a limited number of people are tasked with handling both estimation and implementation functions. Without a strong implementation platform in place, this can lead to problems like limited methodological options (since more advanced methodologies are difficult to implement), rushed or incomplete implementations, very inefficient processes, potentially material

computational errors, and extremely limited transparency or reproducibility around processes.

Problems related to an inadequate implementation platform can cause significant costs to organisations – both in terms of a very inefficient use of scarce resources, but also in the opportunity cost of the lost value that better models could bring them.

With this backdrop of challenges, this article highlights some of the key characteristics of a robust model implementation platform. We feel strongly that investing smartly in this area can pay huge dividends for financial institutions and propel their risk management capabilities forward. In fact, a strong platform can help ameliorate the recurring challenge of limited analytical talent that all risk teams face.

Transparency

An overarching theme of a best-in-class implementation platform is transparency. There should be a central environment where production models can be easily accessed, reviewed and executed. This means that all details around a production modelling system are readily accessible, organised and easily understood. This would include important elements like model parameters, details around the models, code needed to tie models together, and definitions of inputs and outputs into the system. It should also be possible for both technical and non-technical users to execute the models and easily view results across multiple dimensions. In other words, it should not be the case that only the model developer can generate model results.

Reproducibility

Model results should be easily reproducible. This is a cornerstone test for model validation to ensure repeatable accuracy. When multiple analysts re-perform model calculations independently, there should be no material differences in results. This should be the case from data creation (starting with any initial data pulls from source systems) all the way through to the generation of the target output variables. All model implementation testing starts from this foundational concept. Without reproducibility, many hours will be wasted in the process of reconciling results across different model runs.

Consistency of calculations

Key calculations must be consistent across all models, and ideally across the firm. Modularisation of intermediate programmes or calculations can make this easier. To the extent possible, teams across the organisation should adhere to similar implementation standards. There should be standard ways to handle common issues like defining inputs and outputs, looping through time horizons, incorporating economic variables and writing code in parallel. Re-performing the same calculations using fresh code for each implementation increases the risk of errors and unnecessarily lengthens reviews. Make sure changes

to shared files are versioned and controlled to lighten the burden on reviewers and to further decrease risk of errors or miscoding.

Timeliness of results

Model results should be generated in a timely and predictable manner. This means taking care to write efficient code without unnecessary, processing-intensive steps (like writing large data sets to disk without reason). Systems that do not generate results in a timely manner run the risk of missing important submission deadlines. If an error is found in review and model runs need to be re-executed, it is critical that results can be generated quickly. Many best-in-class modelling frameworks are very computationally intensive, necessitating the distributions of calculations across of grid of machines in order to be able to run in a reasonable time frame.

Thorough documentation

Processes should be thoroughly documented. There should be clear procedures for executing models, including detailed descriptions of required inputs and outputs. Parameter settings should be documented and recorded, and the process flow should be clear and easy to follow for all stakeholders. Any code used to create input data sets should be retained and no ad hoc processes should be used to generate inputs for production processes. Visual process flows are helpful and facilitate communication between stakeholders to ensure consistency of expectations.

Documentation requirements also apply to code. Code should not be unnecessarily complex, and should be well-commented to facilitate understanding by reviewers. It should be visually 'clean' and readable by an independent analyst. Since significant amounts of time go into code review during model validation, do not underestimate how much time is saved (and how many errors are avoided) by writing clear, concise and thoroughly documented code.

> **"A strong platform can help ameliorate the recurring challenge of limited analytical talent that all risk teams face."**

Audit trail

It is critically important that production model processes used for financial reporting produce an 'audit trail' – in other words, that the executions are reviewable by external auditors. An audit trail generally consists of proof of execution of all

components of the modelling system, who ran it, when it was run and any inputs that were used. Ideally these would be log files generated through the execution of batch code on a production system. Log files should identify any run-time errors during the execution of the code, and should also contain other informative data such as the number of records processed.

Automation

Processes should be automated as much as possible, especially around hand-offs of critical data elements. Automated processes are much easier to control, and thus are less prone to error and have shorter review times. Automation should also reduce run-times by removing human dependencies – for instance, if you remove manual touchpoints from your model process, you can execute the process end-to-end without any input from users. This allows for computationally-intensive processes to be run overnight or on weekends when computing resources are more readily available.

Access control

All production processes should restrict access to unapproved users. While this necessitates some duplication of effort, only designated users should have access to production systems and backup plans for model execution should be in place to prevent delays due to key person dependencies. This is generally controlled by assigning users to specific work groups and only granting access to the system to those work groups.

Conclusion

If the general guidelines discussed above are followed, your model implementation framework will generate results efficiently, accurately and completely. Review times will be minimised and the processes used to generate your production results will be transparent to all stakeholders, improving communication across lines of business. System maintenance costs will be limited – for instance, implementing quarterly model calibration updates will be seamless as the process will be modularised and well-documented. In a world of ever-increasing model complexity and regulatory scrutiny, these are critical characteristics for all model implementation platforms. R&C

Erik Leaver
Director, Advanced Analytics, Risk Management
SAS Institute, Inc.
T: +1 (919) 531 2566
E: erik.leaver@sas.com

Travis Garner
Solutions Architect, Advanced Analytics, Risk Management
SAS Institute, Inc.
T: +1 (919) 531 1517
E: travis.garner@sas.com