Mo.net
Next Generation Operational Modelling with the Mo.net Platform

January 2024
Revision 14

Background

For years actuaries responsible for operational modelling activities have been wrestling with the need to produce accurate & timely results while at the same time satisfying an increasingly demanding governance & control regime. With regulatory reporting timetables continuing to squeeze the time available to complete ever more granular & transparent reporting across a more diverse range of scenarios, the traditional operational modelling solutions cease to provide the performance, stability, flexibility or control required.

Current operational modelling solutions are typically homebrew / end-user developed wrappers for first generation (now legacy) modelling platforms, which were never designed to deliver the requirements of the operational community. These solutions are usually often quite brittle and commonly create more challenges & issues than they are designed to resolve.

This paper outlines how the Mo.net Operational Modelling Centre provides a best-in-class solution to the operational modelling requirements of a modern insurance company, regardless of its size and whether it runs on-premise, in the cloud, or in a hybrid environment.

The Financial Modelling Lifecycle

In simple terms, the lifecycle of financial models is broadly the same, regardless of organisation size, product range, modelling platform, or territory / regulatory environment.

Figure 1 - Typical Financial Modelling Lifecycle

This lifecycle can be neatly summarised in the figure 1, although the extent to which each step is followed is largely a function of organisation size, maturity and associated risk management framework.

While the focus historically was to develop ever more elaborate modelling solutions to cope with the latest whim of the actuarial / risk management community or the next cycle of regulatory change, this is now shifting towards the need to derive business value / actionable insight from operational modelling on a regular / on-demand basis.

Perhaps the main challenge within the operational modelling arena is ensuring that an appropriate level of governance and evidence is in place for each element of the modelling lifecycle, without constraining the organisation so much that regular cycles of change cannot be efficiently managed. For example, forcing an organisation to proceed around the full change cycle for something as simple as a rate change would be overly bureaucratic & slow. Conversely, allowing small but potentially significant changes to model logic to make their way into the live environment without a suitable level of control & approval would be inappropriate.

The Modelling Paradox

Financial modelling platforms have always tried to balance three, usually competing, requirements:

- Flexibility
- Performance (including scalability)
- Control (including audit, change management and approval)

Traditional platforms have usually been able to provide any two of these requirements but at the cost of the third. Any modern operational modelling platform really needs to support all three requirements without compromise.

What's required is a proportionate level of control & flexibility appropriate to the risks & volatility of the fundamental modelling artefacts, while at the same time maintaining a consistent level of performance and ensuring a suitable level of segregation between the previously blurred lines of development & operational communities.

Operational Modelling Objectives

The fundamental objective of any operational modelling solution can perhaps be distilled into the following question:

> How can I be sure which versions, of which models were run by which users on which date, at what time, with which data, assumptions & parameters to produce which set of results and for what purpose?

While this appears to be a pretty innocuous looking requirement, the challenge associated with delivering this requirement should not be underestimated. In many ways, the main problems stem from the first generation (now legacy) financial modelling tools released in the late 1990s, but still used in most modelling environments today. These solutions were highly proprietary, with

closed architectures and more focused on the job of developing rather than running / controlling models in an operational environment. Any attempts to address emerging operational modelling requirements have usually involved the development of somewhat brittle end-user / wrapper solutions rather than adopting vendor developed & support capability.

Operational Modelling Requirement Domains

There are four primary operational modelling requirement domains as illustrated in Figure 2 below.

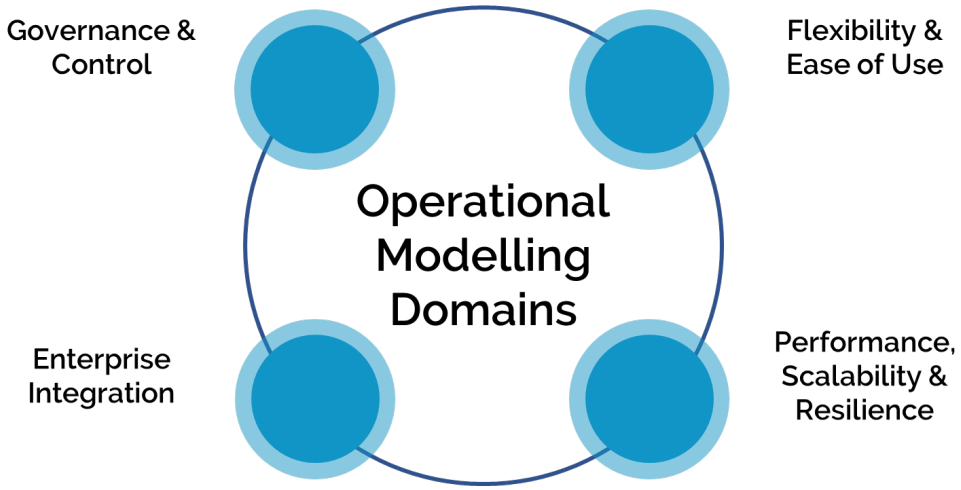

Figure 2 - Operational Modelling Requirement Domains

A summary of the core requirements under each of these domains is provided in Table 1 below.

Governance & Control	- Segregation of development & operational activities, roles, users and permissions - Atomic-level versioning and history of components - All operational modelling artefacts – models, assumptions, data & results – should be locked down to unauthorised change or use - Any authorised change is recorded and subject to 4- or 6-eyes approval - Strict audit of access & action, by user or role, or by Active Directory group membership
Flexibility & Ease of Use	- The operational modelling platform should be accessible to resources who aren't necessarily familiar with the underlying modelling logic - Different modelling artefacts can be combined in order to meet a variety of operational objectives - Unattended operation, scheduling, templates and triggers - Automated "on system" reporting & analysis - Collecting associated modelling artefacts together - Access from any device and any location (subject to corporate access constraints) - Ability to download operational models back into the development environment for change / triage

Enterprise Integration	- The operational modelling platform should integrate with existing enterprise technology components, such as data warehouses, business intelligence solutions, finance systems and infrastructure - Operational modelling solutions should complement existing corporate assets, not necessitate wholesale replacement - Integration points with industry-standard monitoring, alerting and service management systems - Identity & access management with Active Directory
Performance, Scalability & Resilience	- Any platform should provide industrial strength performance & scalability when the operational modelling demands of the business require it - The platform should flex to the operational needs of the organisation, not constrain it - Accelerators in the form of pre-configured runs / templates - Cloud-ready, on or off premise or hybrid implementations - Redundancy / failover of key services to maintain operational availability - Detailed exception handling & reporting to help when things do go wrong

Table 1 - Primary Operational Modelling Requirements

Introducing the Mo.net Operational Modelling Centre

The Mo.net Operational Modelling Centre (the "OMC") is specifically designed to satisfy the operational requirements of any insurance undertaking, while at the same time meeting wider enterprise architecture requirements, such as cloud readiness, cross device access, security, multitenancy, etc.

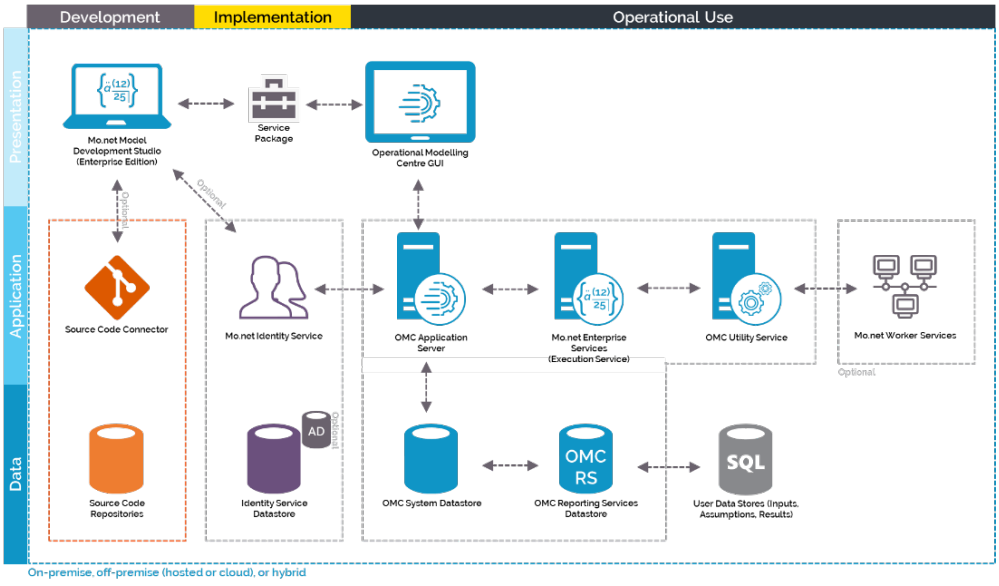

Figure 3 - OMC Component Architecture

The OMC leverages the service-oriented design of the Mo.net platform allowing different service components to be loosely coupled together to meet the specific needs / constraints of the customer, as noted in Figure 3 above. The OMC can be deployed in a traditional on-premise manner, to the cloud, or to variety of hybrid environments, and consumed via a browser on any device.

Delivering the Objectives of Operational Modelling

The OMC sets out to respond to the fundamental operational modelling objective noted above – "How can I be sure which versions of which models were run by which users on which date, at what time, with which data, assumptions & parameters to produce which set of results and for what purpose?"

Models & Versions

The OMC actively prevents model structure or logic (code) being changed in the operational environment. Only data, assumptions and run-time parameters can be changed within the OMC. Once a model has been developed and tested in the Mo.net Model Development Studio, it is published as a service package for use within the OMC (as a task). The specific version of the model (and the underlying Mo.net kernel) is baked-into the service package providing robust & transparent version control and end-to-end audit. Furthermore, if source control is used during model development activity, atomic level change management & evidence is also on offer.

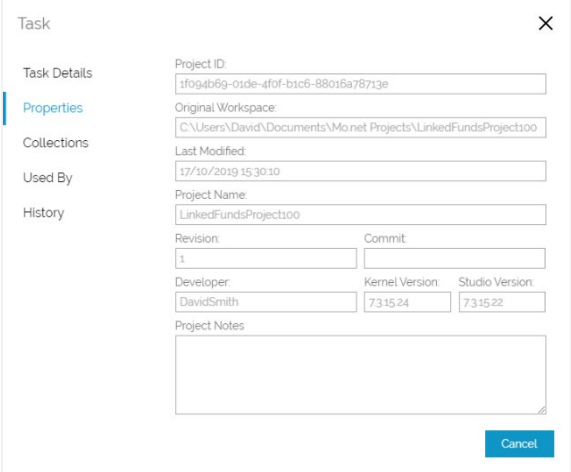

Figure 4 - Importing a Mo.net Service Package into OMC

Who?

Not only does the OMC capture who was responsible for model development, but it also captures who was responsible for bringing the model into the operational modelling environment, and more importantly, who "touched" the model or any other modelling artefact (tables, parameters, datasets, etc) at any point after they are introduced to the OMC. The OMC also captures a complete record of which user was responsible for initiating a modelling job, with or without a 4-6 eyes approval cycle.

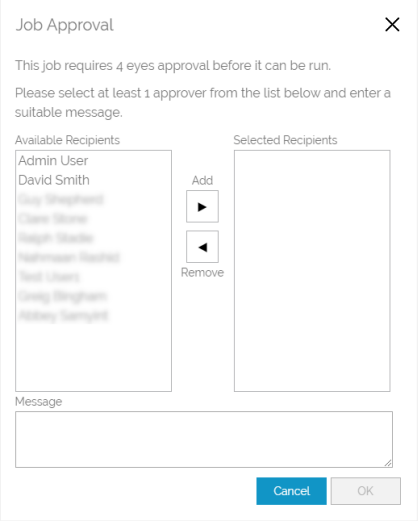

Figure 5 - Requesting Approval to Run a Job

Date & Time

The date & time of every single touchpoint is recorded within the OMC for auditing / reporting / analysis later. This includes successful & unsuccessful system sign ins, sign outs, artefact creation, artefact use, downloads, and change.

Data, Assumptions & Parameters

Obviously making sure the correct version of data, assumptions & parameters are used is as important as ensuring the appropriate version of the underlying model is in play. Model assumptions and reference data, in the form of tables, are managed alongside the models to which they apply. These can be sourced from the same range of diverse sources available across the rest of the Mo.net platform – databases, spreadsheets, text files, real time feeds, etc.

We recognise that assumptions & parameters do change more frequently than models and therefore unlike models, which must be changed within the model development environment, OMC allows these to be changed (under strict control) directly within the system.

The design of OMC also recognises that source data (policy data, market data, schedule data, etc) is often held in stores external to the operational modelling environment and that making copies of large datasets is not desirable (to ensure one version of the truth). As a result, the OMC does not manage source data itself, but instead leaves it at rest in source repositories and instead uses industry-standard checksums to monitor whether data has changed between uses.

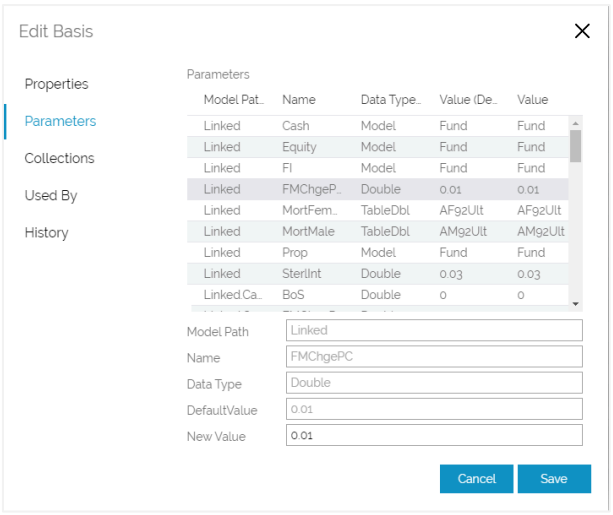

Figure 6 - Editing an Existing Modelling Basis

Results & Output

Since any operational modelling solution only really exists to enable the creation of financial modelling metrics for internal decision support and external reporting, ensuring that results and other output can be delivered in an efficient, flexible and secure manner is a core feature of the OMC. OMC can deliver results to a range of targets – databases, files, etc – or trigger the creation of custom reports & analytics within the platform. This helps avoid moving large volumes of output data away from the OMC platform, which therefore removes some of the control over it.

Why?

All operational modelling artefacts held within the OMC environment include a comprehensive set of metadata. This metadata allows the user community to capture notes about why specific versions of models or assumptions exist, or what the specific version of output is to be used for, or why a specific run has been done. Furthermore, OMC includes a concept of collections to allow all related modelling artefacts – models, data, assumptions – to be grouped together in a logical manner.

Other Features of OMC

While the primary focus of the OMC is obviously to address the fundamental governance & control requirements of operational modelling, the platform does include a wealth of additional features designed to improve operational agility, risk management, and total cost of ownership. Some of these additional features are listed in Table 2 below.

- Collaborative multi-user platform providing a single version of operational modelling artefacts
- Ability to include non-Mo.net tasks, e.g. Windows command line calls and SQL Server Integration Service packages
- Scheduling / delayed start of jobs, perhaps until data or assumptions have been populated into source data stores
- Dashboard of recent changes to artefacts
- Activity monitor showing all live system activity / infrastructure load
- Predicted finish times of jobs
- Flexibility workload distribution mechanisms, using multiple cores or worker services, HPC or combination thereof

- Service management integration & alerting - System dashboards and capacity warnings - External job triggers – service endpoint - Support for multi tenancy (teams or organisations) and strict segregation of artefacts	- Controlled download of assets out of the OMC environment – results, models and assumptions - Bulk import / export of jobs - Agnostic to version of the Mo.net kernel, allowing old jobs to be re-run with the kernel original in use

Table 2 – Additional Features of OMC

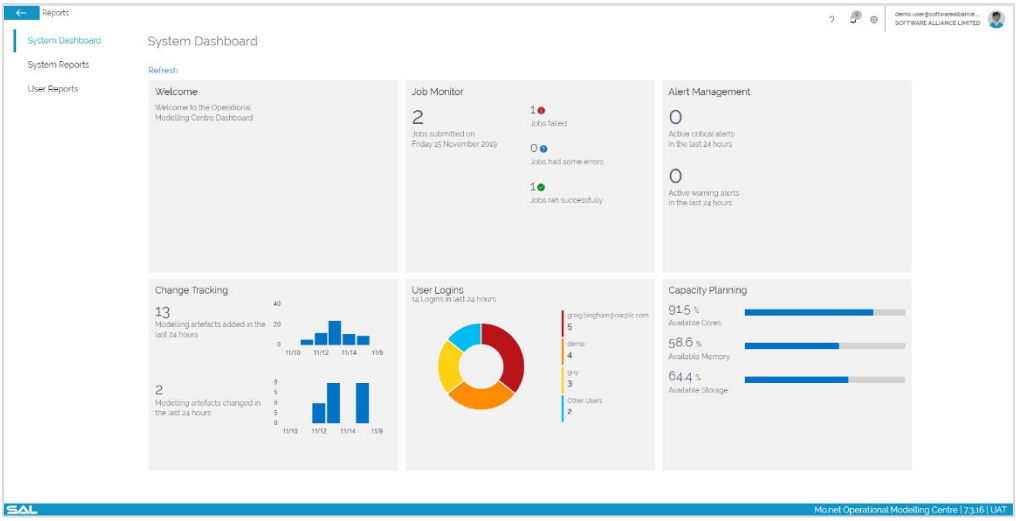

Figure 7 - OMC System Dashboard

Conclusion

In summary, the OMC offers a completely new way of enabling robust, efficient and cost-effective operational modelling activity. It facilitates the formal segregation of development and operational communities and provides the operational users with platform that supports their specific requirements. They no longer have to rely on systems that were designed to support model development or wrappers developed by the end-user community to fill the operational modelling void.

Contact Us

To find out more about the Mo.net Financial Modelling Platform, our range of operational modelling solutions, and to arrange a demonstration of the Operational Modelling Centre, please get in touch.

Software Alliance Limited
30 Stamford Street, London, SE1 9LQ
Tel: +44 (0) 20 3964 2755
www.softwarealliance.net
hello@softwarealliance.net

Author: Guy Shepherd