/\o.net Financial Modelling Platform
Data Integration with Mo.net

March 2017

Background

Enterprise insurance data often comes from a wide range of sources / technologies, in a huge variety of formats / structures, and with varying degrees of quality, completeness and understanding. Unfortunately gaining access to good quality, well understood sources of data appropriate for downstream business applications is not always straightforward. Even when necessary sources of data can be determined, attaching these to financial modelling systems can pose a real challenge, with extracts from source systems into flat text files usually the only reliable (albeit suboptimal) solution.

This short paper outlines some of the typical data challenges associated with financial modelling and highlights how specific features of the Mo.net Financial Modelling Platform enable customers to streamline & safeguard their data integration journey without resorting to additional processes / technology or using legacy integration approaches.

Scope

In the context of this paper data means policy data (varying by member / policy / record / coverage) and assumptions, which are largely fixed within the context of a given projection / modelling run, although parameters may be selected / modified based on the features of the policy data (e.g.. mortality tables modified for smoker status or impaired life features)

The Challenge

Traditional financial modelling platforms impose significant restrictions / constraints on how data can be consumed by user-developed models. These platforms, which were originally developed in the mid-1990s were designed to work with two dimensional, highly structured, good quality data – either in the form of a CSV / delimited or a spreadsheet worksheet file. The assumption was that data could be made available in a form required by the modelling platform / models. Unfortunately the enterprise data held by insurers and required for financial modelling isn't always available in the readily accessible source, format or level of quality, which invariably creates a cottage industry of data integration / remediation. For example, the administration & ledger platforms developed in the 1970s / 80s were never really designed to integrate with other parts of the insurance enterprise, which makes sourcing empirical primary data a particular challenge.

Legacy modelling platforms also struggle to provide integration with non-flat file data sources (holding either policy or assumptions data), which means insurance enterprises often need to develop elaborate data integration / staging area solutions between primary data sources and the financial modelling environment. These data integration steps typically involve consolidating a variety of data

sources into a warehouse or mart that allows huge volumes of data to be transformed from a variety of structures / formats into the format / structure required by the financial modelling platform / model. While there is potential value in implementing these staging areas, the time, cost & effort associated with their development is not insignificant.

Note: Some of the financial modelling tools do provide additional tools to help with transformation / aggregation of source data, but these come at additional cost and often with further functional / non-functional constraints.

Clearly the fragmentation of the data integration process can have a significant impact on operational activities. Even when source data is finally available, rework is often required to fix up / remediate data inconsistencies / gaps that fail at the modelling stage. This can lead to significant delays in modelling – it can be days or even weeks before source data is in a fit state for operational use.

Data Quality

Any data integration environment also needs to concern itself with the remediation of poor quality data, since downstream models are generally very fragile when it comes to missing / incorrect data elements. Even worse they may not raise an exception at all and process data as if it was complete, leading to volatile or incorrect results.

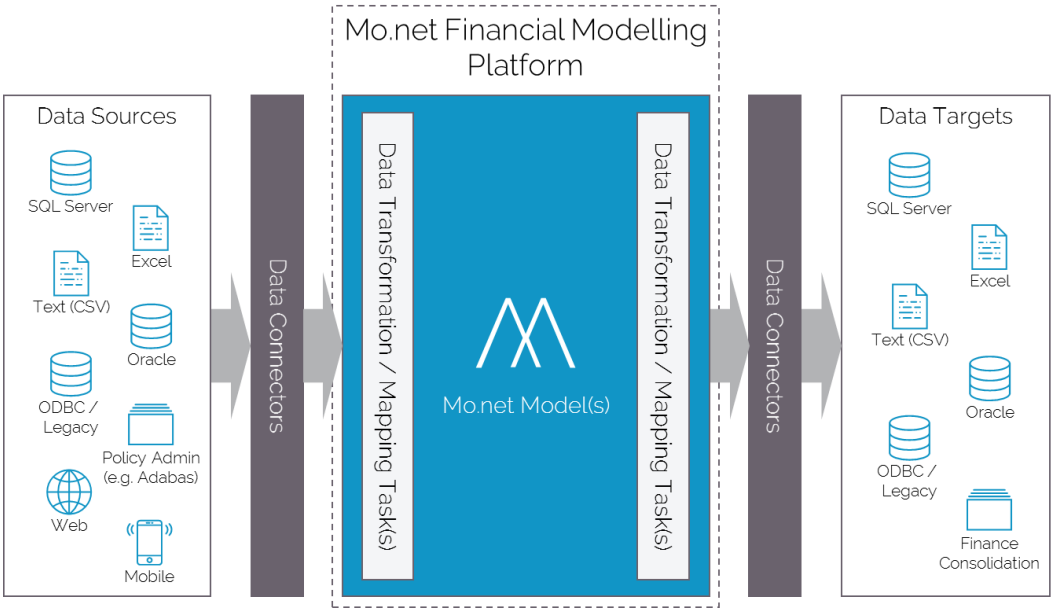

Figure 1 - The End-to-End Modelling Data Journey

And the problem doesn't end when source data finally enters the modelling environment. Model output / results are equally challenging for traditional modelling platforms to handle, with results typically being deposited into text or proprietary files which then require further data integration / transformation activities to allow them to be used in downstream analysis / reporting work. This extends the end-to-end financial reporting timeline still further, and increases the risk of failure due to the sheer volume & variety of dependencies / handoffs.

Legacy Data Sources

Before the advent of commercially available relational database management systems (RDBMS), insurers and administration system providers utilised a range of sophisticated but relatively opaque methods for storing data, usually on mainframe systems. These storage techniques often "packed" the data to make the best use of available storage / memory or held the data in proprietary hierarchical, dimensional or entity-relationship formats. Unfortunately these techniques have traditionally made the data very difficult to "read" without using expensive extract-transform-load (ETL) solutions. Furthermore, knowledge of these systems / formats is starting to erode as the original developers / engineers leave the active workforce.

Our Approach

Mo.net has been specifically designed to work with a range of traditional & more modern / industrial data sources hosting almost any structure / format of data content. This allows models to be aligned to product features / data rather than having to make data fit with a specific modelling platform or model structure. Mo.net does not have a prescriptive data model / schema; the data required by a model evolves in line with the development of the model itself. Even that structure doesn't have to be fixed, as through more advanced data transformation & modelling techniques the features of the model / the calculations themselves can be dynamically created based on attributes or values in the data.

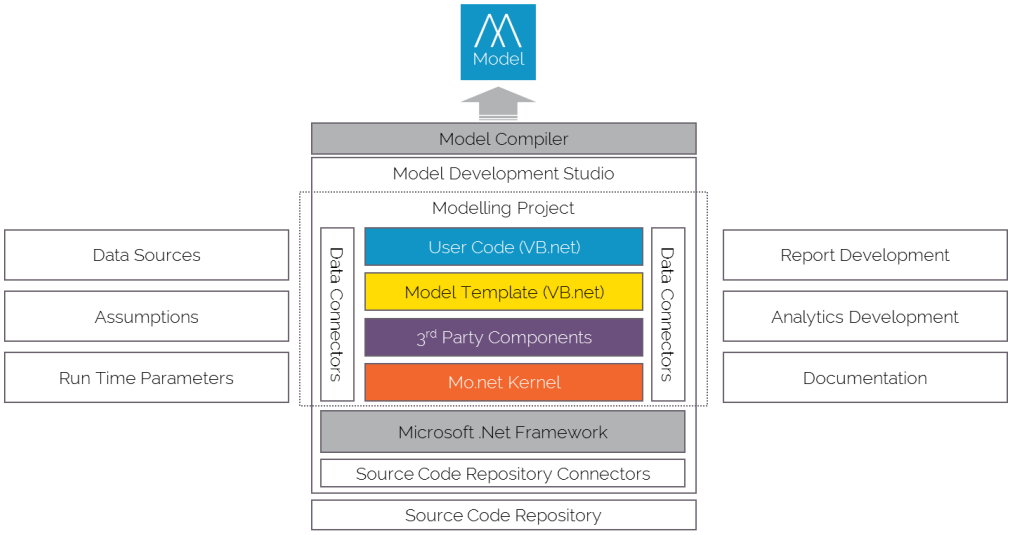

Figure 2 - Mo.net Data Connectivity During Model Development

Furthermore, unlike most other financial modelling platforms, Mo.net provides a clear layer of abstraction between the model logic and specific data & assumptions. This means the same model can work with range of different data sources / structures with little or no development work. Modelling features can also be driven by data elements / content, for example if rider benefits exist in some policies but not others, the model can create / use associated rider modelling components "on the fly".

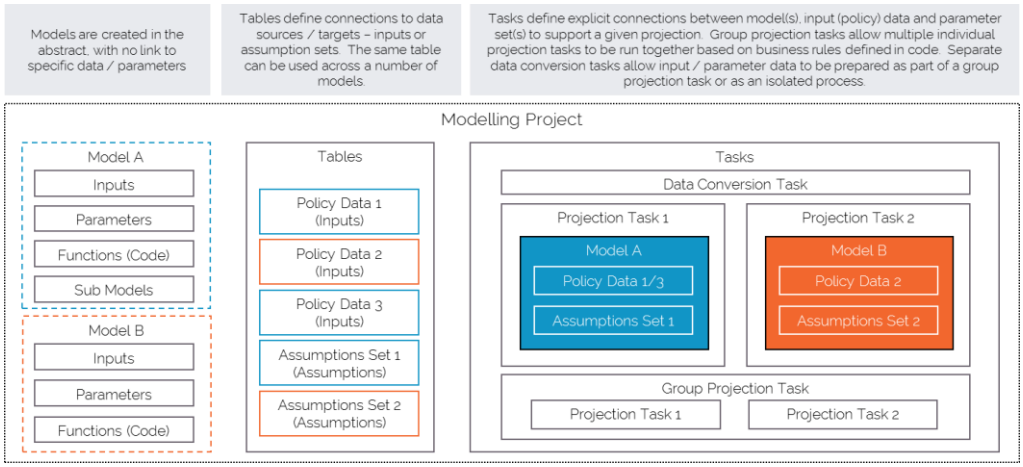

Figure 3 - The Mo.net Paradigm

Mo.net can integrate directly with a range of data source providers – either primitive sources (limited only by available ODBC drivers), data warehouse /

integration environments (using native or ODBC connectors), or traditional text / spreadsheet files.

Once a connection to a data source / data integration environment (or legacy flat file) has been established, data conversion tasks can process the data in situ, regardless of the format / structure, and inject the data directly into the model. These data conversion tasks are developed inside the modelling environment itself – using the same interface / language (VB.net) as the model. This provides almost limitless data transformation flexibility, free from the constraints / limitations of any separate tool, and using an environment which model developers will be familiar with.

Once source data / assumptions have been consumed by a model, results can be inserted into the same DBMS (or legacy data store) environment as the source – allowing source data / results to be aligned, or a separate dedicated results store for analysis / reporting / publication. The integration flexibility of Mo.net is only limited by the availability of drivers to connect to these target data stores. Any element of the model results (or the intermediate calculations) can be preserved and written to any target output environment.

Finally, we are currently working to allow any attribute of the data source / dataset to be preserved through the model and made available in the output / results. This will provide granular model look-through capability now being demanded by regulators and insurers alike, to ensure that results can be aligned to primitive sources of data / assumptions.

Further Information

For further information regarding any aspect of the Mo.net Financial Modelling Platform and its data integration capability, please contact Software Alliance Limited.


Software Alliance Limited
Adam House, 7-10 Adam Street, The Strand, London, WC2N 6AA


+44 (0)207 520 9477


www.softwarealliance.net


hello@softwarealliance.net

Author: Guy Shepherd