

## Topic 4d – Architecture of Oasis

[Mark Pinkerton] I'm going to talk about the Oasis architecture. When we were designing the Oasis architecture, there were three goals that we needed to keep in mind. The first was the ability to support any model on any methodology and replicate the results. This can be achieved either through using the existing Oasis components and deploying model-specific data or by creating custom components for a particular model and deploying these.

The second goal was deployability. The Oasis system needs to be deployed to a range of environments from enterprise risk systems at insurance companies or reinsurance companies for use by underwriters or risk managers through to model development toolkits for use by scientists and academics. The third goal, then, was high performance. The system needs to be able to run industrial strength models against large portfolio of risks and acceptable runtime. And that runtime is timed answer, which includes not just the model execution time, but the time to upload the exposure data and to retrieve the results.

So the Oasis architecture is a four-tier architecture. The source code for the system is available on GitHub a generally open-source with a permissive license. The first tier is the flamingo user interface. This is a reference UI for common catastrophe modelling workflows. The user interface is implemented in R shiny, which is based on top of the popular statistical language R and provides a lot of configurability in terms of different analytics and data visualizations.

The second tier is the Flamingo server. This is a SQL Server that provides configurable catastrophe modelling workflows and also holds exposure data in Oasis' canonical data format. It also provides data transformations into an array of the canonical data format from common market formats. The third tier, then, is the Oasis mid-tier. This is a RESTful API layer that provides web services that enable you to upload model data, upload exposure, configure and run analyses, and retrieve results. It also provides security and user authentication.

The heart of the system, then, is Ktools, or kernel tools. This is a high-performance layer for multi-process, model execution, financial modelling, and analytics. It is written in c++, provides a set of components that are reference components for catastrophe modelling, and can also be run directly from the Unix command line.

So there are three main ways in which the Oasis system can be deployed. The first is a standalone catastrophe modelling platform. In this mode, all four tiers of the system are deployed. In the second mode, we deploy the mid-tier and the calculation back-end. This provides an API for catastrophe modelling and is appropriate if there's an existing application which requires catastrophe modelling analytics to be embedded.

In the third mode, we just make use of the calculation back-end. In this mode, the modelling analytics can be ran directly from the command line or embedded in another program. This mode is most appropriate for PAR analysts, model developers, scientists, or academics.