Benefits of a Cloud Environment: An Example of Operational Risk Capital Model

Matlab Computation Finance Conference
24 May 2018

Giles Spungin, PhD(Lon)
Global Head of Regulatory Compliance and Operational Risk Analytics
## Agenda

<table>
<thead>
<tr>
<th>Topic</th>
<th>Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model Overview</td>
<td>1</td>
</tr>
<tr>
<td>Transition to the Cloud</td>
<td>2</td>
</tr>
<tr>
<td>Life in the Cloud</td>
<td>3</td>
</tr>
</tbody>
</table>
Model overview
Model Overview - Overview of the Operational Risk capital model

**Background**
- Model consists of three sub components, which each consist of up to 72 units of measure
- For each unit of measure a Monte-Carlo simulation (1 million simulations) is run. (72*3*1M = 216M simulations), 30 hours compute time on a standard desktop machine
- End to end process takes approximately 6 weeks to run due to manual processing pre- and post-model execution

**Data Inputs**
- Data:
  - External
  - Internal
- Scenarios
- Business indicators
- Data Report

**Parameterisation**
- a) LDA:
  - External Component
  - Internal Component
- b) SAM:
  - Scenario Component

**Weightings of models**
- Simulation and Aggregation:
  - Loss Component (internal + external)
  - Combined model

**Capital estimation and Allocation**
- Economic Capital Calculation for the legal entities in scope:
  - Aggregation at legal entity (correlation)
  - Economic Capital (99.9%-EL)
- Economic Capital for HSBC
  - Adjustment for out of model scope entities

Process at scenario Level
Process at Unit of Measure Level
Process at Group Level
Model Overview - Stakeholder requirements and challenges

HSBC is adopting a Cloud First Strategy, in order to benefit from:

- Capability – leverage Cloud Service Provider (CSP) technology offerings
- Adopt a flexible consumption model
- Increased Delivery Flexibility – dynamic scaling of immediately available capacity

<table>
<thead>
<tr>
<th>Stakeholder requirement</th>
<th>Legacy implementation</th>
<th>MATLAB Cloud implementation (new)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Run the model more frequently</td>
<td>Workbooks are slow and manual</td>
<td>MATLAB parallel computing toolbox &amp; MDCS* enables faster computation</td>
</tr>
<tr>
<td>2. Expand the model to more regions</td>
<td>Infrastructure is not readily scalable</td>
<td>Cloud solution with MDCS* scales dynamically</td>
</tr>
<tr>
<td>3. Access to model globally</td>
<td>Model can only be run from a desktop in one location</td>
<td>Cloud is globally accessible</td>
</tr>
<tr>
<td>4. Streamline the development cycle</td>
<td>Complex workbook structure</td>
<td>MATLAB toolboxes, integration with GitHub and computer science tools, e.g. code profiling tool and unit testing framework, speeds up development cycle</td>
</tr>
<tr>
<td>5. Reporting is seamless and timely</td>
<td>Outputs to PDF and Excel</td>
<td>MATLAB reporting generator automates reporting</td>
</tr>
</tbody>
</table>

* MDCS – MATLAB Distributed Computing Server
Model Overview – Legacy implementation and improvement opportunities

1. Legacy model execution is manual and requires additional validation steps during the process
2. Automation of data ingestion in the current implementation is difficult and lacks scale
3. MATLAB data types and toolboxes allow for easier automation of data ingestion and manipulation
4. MATLAB parallel computing toolbox speeds the cloud implementation – requires additional coding to implement
5. MATALB report generator speeds automation of report generation
Transition to the cloud
Transition to the Cloud – Approach Overview

Phased approach, while building the cloud environment in parallel:

1. Translate from legacy implementation to MATLAB, in serial
   - Functional programming design and encapsulation of functions, which enables code reuse
   - Use of inbuilt MATLAB profiler to assist in vectorizing code and streamlining performance
   - Use of inbuilt MATLAB unit testing framework to formalise and automate testing
   - Refactoring of code highlights areas that will benefit from parallelisation

2. Refactor and vectorize serial code, to streamline performance
   - Development of parallelised code in the parallel computing toolbox speeds development cycle
   - Development scaling is representative of the MDCS* in the cloud

3. Parallelise the code to improve performance further
   - Development with MATLAB parallel computing toolbox enables cloud build as a separate work stream

4. Build cloud environment as separate work-stream
   - Development with MATLAB parallel computing toolbox enables cloud build as a separate work stream

* MDCS – MATLAB Distributed Computing Server
Transition to the Cloud – Performance increase at each phase

- Memory limitations of legacy approach removed by moving to MATLAB
- Largest increase in speed by refactoring and vectorizing code
- Parallelisation greatly speeds calculations and aggregation of results
Transition to the Cloud – Functional Cloud Design

Cloud design

1. **Portability**: Built on Cloud and MATLAB implementation is portable with minor changes

2. **Usability**: Model developers and model users can access the Cloud using single sign-on process

3. **Security**: Virtual private Cloud, hardened to HSBC standards.

4. **Repeatability**: Integrated with Code storage solutions

Future Proofing

1. **Scalability**: Elastic compute resources

2. **Advanced analytics**: Additional services added easily e.g. Tableau server and machine learning capabilities

3. **User management**: Centralised user management

4. **Automation**: Standardisation of reporting formats enables greater automation
Life in the Cloud
Life in the Cloud – Benefits of parallelisation

- 30 hours compute time
  - Single user
  - One location

- 10 hours compute time
  - Multiple users, many code versions
  - Several locations

- <1 hour compute time
  - Multiple users, single source code
  - Global on demand usage

Parallel toolbox

Desktop

Today
Life in the Cloud – Operational Risk Analytics

Lessons learned

1. **Testing**: A well defined testing plan saves time

2. **Data Security**: Hosting data externally in the cloud is a significant step

Immediate Benefits

1. **Scalability**: Run multiple instances of the model concurrently, speeding delivery of analytics

2. **Usability**: Accessible to resources outside London – fits the global team

3. **Security**: Increased system availability and redundancy

4. **Repeatability**: Centralised code and compute resources allows analysts to separate the design and production environments, reducing possibility of errors

Secondary Benefits

1. **Reporting**: Output visualisation in Tableau

2. **Automation**: Self-service on-demand capabilities

3. **Analytics**: Concentrate on delivering value-add analytics
Together we thrive