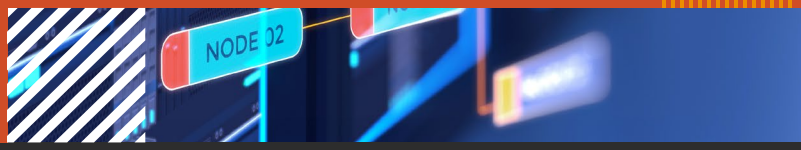
Key considerations and best practices for financial modelling infrastructure



In recent years, models used in financial institutions have undergone a significant transformation. There has been a substantial increase in the number of models across domains, making them an integral part of the risk management and decision-making processes. This proliferation of models is driven by a confluence of factors, including regulatory requirements, technological advancements and increasing complexity of financial markets. Therefore, the modelling infrastructure used for implementing these financial models should also keep up with these evolving changes in the financial markets.

A robust modelling infrastructure should efficiently support financial models across the model lifecycle including model development, deployment, ongoing monitoring and governance. In this article, we will discuss some of the key infrastructure considerations for financial modelling as follows:

- Data management
- Model testing environments
- Support for diverse programming languages
- Support for integration of third-party/ vendor models and solutions
- Efficient allocation of computational power for models and scalability
- Monitoring and performance metrics

pwc

01 Data management

Data management plays a crucial role in the modelling infrastructure as it is central to the effectiveness, accuracy and reliability of the models. Data serves as the foundation across all the areas of the model lifecycle, including model development, validation and ongoing monitoring. The following components could be considered for setting up an effective and efficient infrastructure:

Component	Key considerations
Data discovery, storage and accessibility	Data used in modelling should be appropriately identified, efficiently retrieved, accurately processed and updated. To ensure this, a reliable and secure data storage infrastructure with efficient data retrieval mechanisms must be established to enable timely access for model users. The data used in the financial institution could be structured or unstructured; hence, the modelling infrastructure should have capabilities to support all types of data. This helps in creating a comprehensive data repository, providing access to both historical and real-time data.
Data integration	Data used in modelling is not always internal to the financial institution and is often outsourced from external data providers – for example, market feeds, economic indicator data, bureau data etc. Hence, it is important to implement a comprehensive approach to achieve effective support for external data feeds in the overall modelling infrastructure. This can be achieved by establishing application programming interface (API) connections with the external data providers, as APIs provide seamless and standardised data exchange between external sources and the internal infrastructure. These APIs could also be automated to fetch real-time data. All data sources used in the modelling might not have the same data structures, formats and naming conventions. Therefore, it is crucial to ensure that effective data mapping mechanisms are in place to ensure that data from all sources is accurately processed and efficiently integrated.
Data quality management	Sufficient data quality assurance mechanisms including validation checks, data-cleansing processes and error-handling procedures must be established to ensure that all the internal and external data feeds are accurate, complete and consistent.
Data access control/ security	Financial institutions must adhere to regulatory requirements and ensure that data is handled securely. Sufficient data controls and robust security measures including encryption, access control systems must be put in place for both internal and external data to safeguard sensitive information and prevent unauthorised access. Additional controls should be implemented for external data by establishing secure channels with external data providers and using encryption during data transfer. For example, all market data used for yield curve construction and pricing is sourced from external data providers. Secure encrypted channels should be set up to download this external data, which could be stored in a market data manager (MDM), with sufficient access controls to prevent unauthorised usage and tampering. The infrastructure should be reviewed on a regular basis to assess and patch any security vulnerabilities that occur. Sufficient backup and recovery procedures should be implemented for critical data feeds to avoid potential disruptions and data replication mechanisms should be established to take regular backups of critical data feeds.
Control and governance	It is essential to ensure that the data used is reliable, accurate, secure, well-defined and fit-for-purpose for risk modelling. To achieve this, appropriate and well-defined controls and data governance procedures must be established in place, with clear ownership assigned. Sufficient data procedures should be established for data cleansing, sampling, representativeness, validation and reconciliation. Adequate data quality controls and governance processes should be incorporated to maintain the accuracy and integrity of the data used in the risk models. Data that contains personally identifiable information (PII) must have restricted use cases and be appropriately authorised. There must be clear and established controls to ensure that restricted variables like caste, colour, territories and gender are not used for modelling purposes.

02 Support for diverse programming languages

There are various types of models used in a financial institution, each with its unique characteristics and infrastructure requirements. Hence, the modelling infrastructure should be developed and adopted considering these rapidly evolving requirements in the financial model landscape. Support for diverse programming languages is one of these important infrastructure considerations. Modelling infrastructure supporting diverse programming languages would offer the below benefits:

a) Model developers would have the flexibility to choose a programming language that best suits the needs of a particular modelling solution. For models relying heavily on statistics, programming languages like SAS, Python, MATLAB and R might be more suitable whereas for models where low latency is a key requirement, C++ might be preferred.

b) Various programming languages might have been used in the existing applications and systems of a financial institution. Therefore, a modelling infrastructure that supports diverse programming languages will ensure smooth integration with the existing applications and systems, enhancing overall interoperability.

c) Model developers could also make efficient use of specific frameworks and libraries offered by various programming languages based on their modelling requirements.

The following points could be considered for implementing an interoperable environment for financial modelling:

- A modular architecture could be adopted for the financial modelling infrastructure, providing the advantage of developing different modules in different programming languages. For example, a microservice architecture could be used while developing financial models wherein independent modules in a model could be encapsulated as independent services that can communicate and integrate with each other seamlessly.
- A language-agnostic modelling framework could be adopted to accommodate modules developed in multiple programming languages.
- Set up detailed guidelines on the coding standards to be followed during development to ensure consistency across all the programming languages,
- Ensure that efficient communication protocol mechanisms are put in place for information exchange between various modules of a model that are developed using multiple programming languages.
- Include support to integrate external systems or solutions that might have been developed in multiple programming languages seamlessly in the modelling infrastructure.
- Sufficient testing procedures and mechanisms should be put in place to assess the overall implementation accuracy of the model developed using modules written in multiple programming languages.
- Provide detailed technical documentation regarding the development procedures to be followed.
- Conduct sufficient testing to assess the compatibility and accuracy of various modules of a model written in multiple programming languages. Testing activities should include integration testing, system testing and validation against all general and edge cases to ensure accurate implementation of financial models.

03 Efficient allocation of computational power for models and scalability

Given the significant rise in the number of models within financial institutions, computational power and scalability of the modelling infrastructure plays a crucial role in ensuring effective model implementation and usage. Below are some of the key benefits of implementing a scalable modelling infrastructure with efficient allocation of computational power:

a) Financial models are constantly becoming more complex due to large volumes of data, complex algorithms and advanced methodologies. Modelling infrastructure with efficient allocation of computational power ensures that the models are implemented and executed effectively.

b) As the number of models increases, it becomes crucial to implement a scalable infrastructure that can accommodate the growing model inventory to handle multiple models simultaneously without compromising on performance or responsiveness.

c) Scenario and sensitivity analyses are performed as part of model testing, which involve generating model outputs under different scenarios. In order to run these frequent simulations for many models effectively, sufficient computational power should be allocated.

d) The workload and resource requirements of the modelling infrastructure significantly rise during periods of increased activity such as regulatory reporting timelines. Hence, modelling infrastructure with efficient allocation of computational power would help in preventing bottlenecks and ensuring uninterrupted model executions.

Below are some ways that enable the development of a scalable infrastructure with efficient allocation of computational power:

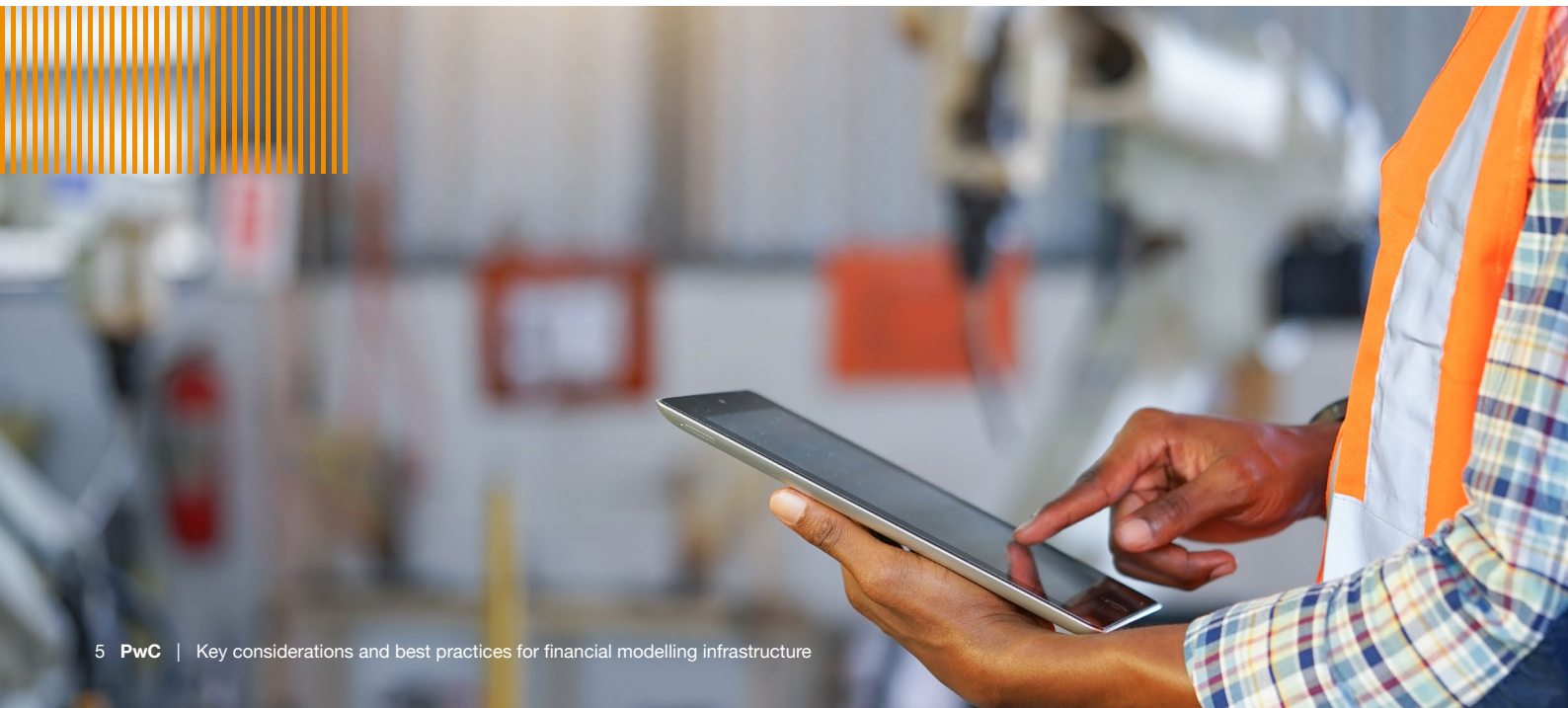
- Perform a detailed assessment of the modelling infrastructure on a regular basis to identify potential bottlenecks affecting performance and areas that need scalability improvements.
- Consider implementing high-performance computing solutions and distributed computing models to process complex model assessments at high speeds across multiple nodes in parallel. This would also help in preventing any single server from overloading, thus increasing the efficiency.
- Introduce graphic processing units (GPUs) for efficient parallel processing. Set up sufficient RAM and fast storage solutions to reduce latency.
- Use cloud computing services which would allow to scale the modelling infrastructure on demand when required.
- During model development, efficient algorithms should be implemented to make use of the computational power available in the modelling infrastructure. A periodical review of the existing model methodologies should also be done to consider implementing simple methodologies without giving up on performance.
- Set up efficient caching mechanisms to store frequently utilised data or computations which would limit repetitive computations and improve response times.
- Optimise network communication between various components of the modelling infrastructure by minimising network latency as efficient data transfer is important for scalability.
- Regularly conduct load testing to simulate heavy workloads to identify and address any potential performance bottlenecks in the infrastructure. Implement feedback mechanisms for continuous improvement based on feedback, monitoring data and evolving requirements.
- Incorporate sufficient monitoring tools and process checks to keep a track of the system performance and analyse resource utilisation patterns to identify areas that should be optimised to improve the overall efficiency.

04 Model testing environments

Financial models often rely on historical data, statistical methods and several assumptions to compute the required risk metrics. So, uncertainties are inherent in these financial models. Hence, these models must be sufficiently tested and validated before deploying them in the production environment. To ensure accuracy and continued reliability of the model outputs, a robust model testing framework should be implemented.

Below are some of the key aspects that should be considered for setting up effective model testing environments:

- Set up a dedicated testing environment that mirrors the production environment as closely as possible to ensure that test results are representative of actual system behaviour.
- Ensure that the production data feeds are accurately replicated in the testing environment to ascertain that the data used in testing is realistic and reflects the characteristics of actual usage.
- Sufficient measures should be put in place to ensure strict isolation between the production and testing environments to prevent any unintended impact on live operations.
- Design and execute sufficient model validation tests including scenario analysis, benchmarking, stress testing and back testing to assess the accuracy and reliability of the models in the testing environment and compare model outputs with expected results.
- Conduct regression testing to verify that changes to any model components, algorithms or the infrastructure do not introduce new issues or negatively impact existing functionalities.
- Develop automated testing scripts and procedures for performing repetitive and routine tests which could be used to validate models regularly and respond to changes in data or market conditions.
- Continuously monitor and evaluate the effectiveness of the testing environment by implementing sufficient monitoring tools to track system performance, response times and resource utilisation to detect anomalies and potential bottlenecks to address them in time.
- Keep track of the changes made in the production environment by implementing version control and ensure that the testing environment is also up to date with the latest versions of models, algorithms etc.
- Ensure that testing environments are compliant with the regulatory requirements and internal governance policies including data privacy regulations, model validation standards and audit trails.
- Maintain comprehensive documentation of the testing environment including the details of the test cases, data sources, configurations and any deviations made from the production environment.



05 Support for integration of third-party/vendor models and solutions

Not all modelling solutions are developed in-house, so financial institutions often rely on third-party/vendor models and solutions for some of their requirements. Thus, it is important to have support for these models and solutions in the infrastructure to enhance the overall functionality, efficiency and flexibility. Some of the crucial aspects that should be considered when incorporating support for third-party/vendor models and solutions are as follows:

Data governance controls: Some vendor solutions are offered only as an online (cloud-based) solution instead of local (on-premises) solution. Moreover, the internal data of the bank might be processed on the vendor servers as part of the model execution process. So, it is vital to ensure that sufficient data governance controls are put in place for vendor models so that internal data is handled appropriately and in line with the bank's privacy and governance policies.

Connection types: There are several types of connections using which vendor models and solutions could be integrated in the financial institution's modelling infrastructure such as API integration, data feeds, manual or automated file transfer, database integration, web services, middleware integration, cloud-based integration, file transfer protocol (FTP) and direct server connections. The choice of the connection type should be made by critically assessing factors like data sensitivity, real-time requirements, security measures, and capabilities of internal financial institution and vendor infrastructure.

Selection of vendor model/solution: Before choosing a third-party vendor model/solution, thorough assessment should be made across the factors including reliability, reputation, infrastructure capabilities, security etc. Assessment could be performed on the following factors:

i. vendor's track record for reliability, support and ongoing updates, customer reviews and responsiveness to addressing security vulnerabilities

ii. scalability of the third-party vendor solutions to ensure that they are able to handle the data volumes, computational demands and user loads as per requirement

iii. security protocols and compliance standards of third-party applications to ensure that they align with the security measures and regulatory requirements of the financial institution's infrastructure

iv. assessing whether the vendor models could be effectively integrated in the existing modelling infrastructure by performing a detailed analysis to identify any dependencies to be managed

v. reviewing whether the customisation options provided by the vendor solution meet the identified modelling requirements and also any future modelling requirements that may arise

vi. assessing the level of support services provided by the vendors and establishing service level agreements outlining the response times, issue resolution and ongoing maintenance.

Testing: Perform rigorous testing including functional, performance, user acceptance testing etc., before deploying any vendor solution in the production environment to identify and address any issues.

Monitoring: Establish monitoring processes to track data accessed by these vendor models and if these models are modifying any of the internal data to ensure transparency.

Documentation and training: Provide a comprehensive document detailing the vendor modelling solution and adequate training material for all relevant personnel on the intranet website for quick reference.

Exit strategy: Develop an exit strategy in case there is a need to replace or discontinue the use of a third-party model/solution and ensure that data can be migrated seamlessly, and dependencies can be managed effectively.

06 Monitoring and performance metrics

The performance of the modelling infrastructure should be reviewed regularly to ensure that all the models implemented on the platform are operating optimally. This can be accomplished by implementing a comprehensive monitoring and performance measurement strategy using key performance indicators (KPIs) and alert management systems to proactively address any potential concerns that affect the infrastructure. These KPIs should also be reviewed regularly to assess if any additional improvements could be made to ensure efficient detection of performance changes.

Some of the important KPIs that could be considered for assessing the performance of the infrastructure are provided in the table below:

KPI	Metric	Description
Resource utilisation	CPU utilisation, memory usage, disk I/O and network throughput	Monitor the use of computational resources regularly to ensure there is sufficient capacity and resources available on the infrastructure for running the models efficiently. This monitoring metric also helps in identifying any potential bottlenecks to optimally allocate the resources.
Response time	Model execution time	Measure the system's response time to assess how quickly models can process data and provide the outputs. Optimise infrastructure to meet performance targets and address latency concerns, especially in real-time risk management scenarios.
Error rates	Error rate per model	During model execution, there might be possibilities of occurrence of errors, and it is crucial to track these occurrences and perform a root-cause analysis. A rising error rate may indicate issues with data quality, model performance or infrastructure stability. Early detection of these errors would help in preventing inaccurate risk assessments.
Scalability metrics	System performance under increasing load	There would be peak times when the workload on the infrastructure would increase significantly – for example, during quarterly or annual regulatory reporting periods. So, it is crucial to evaluate the infrastructure's scalability by assessing its performance during these times and ensure that the optimal maximum capacity is available.
Data ingestion and processing	Data ingestion rates, data processing time	Analyse the efficiency of data ingestion and processing pipelines to ensure that the infrastructure can handle all the incoming data volumes, process it in a timely manner and deliver accurate inputs to the models for processing.
Latency and lag metrics	Latency in data processing, model execution times/lag	The execution time for generating the model output once input is provided should be assessed regularly to identify any latency concerns. Since low latency is crucial for real-time risk management models, this assessment would help to identify and address any delays in the model execution.
Infrastructure availability	Uptime percentage, downtime percentage and mean time to recovery	It is important to track the availability of the infrastructure to ensure that it meets all operational requirements. All downtime events should be thoroughly monitored and assessed to implement sufficient measures to minimise disruptions, which could be potentially achieved by setting up redundancy and failover mechanisms.
Resource forecasting and capacity planning	Predictive analysis for resource needs; resource usage trends and growth rates	The infrastructure resource requirements should be assessed and forecasted on a regular basis to prudently scale based on the predicted workloads to ensure optimal performance and prevent disruptions. To determine effective strategies for capacity scaling, historical performance data along with the trends in resource usage could be assessed to accommodate the increasing demands of modelling requirements.
Security metrics	Security incident rates, unauthorised access attempts	To protect sensitive data and models on the infrastructure, it is important to ensure that sufficient security measures and protocols are implemented, also considering the regulatory requirements related to the data privacy and security guidelines. The performance of these security measures and protocols should be assessed regularly to ensure their effectiveness. This could be achieved by monitoring security incident rates, unauthorised access attempts etc., to identify any potential threats or breaches.

The infrastructure areas discussed in this article are not exhaustive but cover the key considerations that would help financial institutions to build an effective modelling infrastructure. A few other key considerations are given below:

Model versioning: Financial models undergo continuous updates due to address changes in market conditions, regulatory requirements and evolving business strategies. To ensure that different versions of these models are effectively managed, tested and deployed in a controlled manner, model versioning and a well-defined deployment process (release management) must be implemented. Model versioning helps in traceability, auditability, collaboration and compliance.

Collaboration tools and workflow management: Implementation of efficient collaboration tools and workflow management systems are key to facilitate effective communication and coordination among team members involved in the modelling. This includes version control systems, project management tools and communication platforms. For example, to address any identified findings or issues, model development teams can utilise the workflow tool to provide remediation plans to model validation teams. This can involve documentation and outlining actions to rectify the identified problems and discrepancies.

Documentation: Detailed documentation of the infrastructure, including configurations, dependencies, integration, controls and governance policies and procedures should be maintained along with the business requirements document (BRD), functional specification document (FSD) etc. This ensures that the knowledge about the infrastructure is well-documented and transferable.

Conclusion

Establishing a robust financial modelling infrastructure is crucial for banks to navigate the ever-evolving financial landscape. The considerations discussed above, ranging from data, flexibility of programming languages to the computational power and scalability emphasise the need for a dynamic and adaptable modelling infrastructure. Therefore, financial institutions should actively embrace the emerging technological advancements, while keeping up with the regulatory changes and industry best practices, positioning themselves at the forefront of financial modelling.

About PwC

At PwC, our purpose is to build trust in society and solve important problems. We're a network of firms in 151 countries with over 360,000 people who are committed to delivering quality in assurance, advisory and tax services. Find out more and tell us what matters to you by visiting us at www.pwc.com.

PwC refers to the PwC network and/or one or more of its member firms, each of which is a separate legal entity. Please see www.pwc.com/structure for further details.

Contact us

Manish Maini

Partner – Risk Consulting
PwC India
manish.maini@pwc.com

Vikash Jaiswal

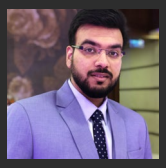
Director – Risk Consulting
PwC India
vikash.jaiswal@pwc.com

Ajay Kumar Gupta

Director – Risk Consulting
PwC India
ajay.kumar.g.gupta@pwc.com

Shreyans Ranka

Senior Manager – Risk Consulting
PwC India
shreyans.ranka@pwc.com

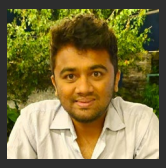


Dwaipayan Majumder

Senior Manager – Risk Consulting
PwC India
dwaipayan.m.majumder@pwc.com

Venkata Subba Bharath Kumar Miriyala

Senior Associate – Risk Consulting
PwC India
venkata.subba.bharath.kumar.miriyala@pwc.com



Akash Anand

Senior Associate – Risk Consulting
PwC India
akash.a.anand@pwc.com

pwc.in

SG/March 2024-M&C 35834