

CLIENT OBJECTIVES' SUMMARY

Client Background

A London-based company, this ebusinessware client has developed high-performance, high-volume electronic trading, decision-support and risk management solutions for the world's premium investment banks.

The client has a financial application, a framework, which provides a rich set of Fixed Income, Futures, Options and Risk Calculations. The application was developed in C++ environment that was the major hindrance for it to be used from the remote location or from web based application. A JAVA framework was to be developed that would act as a middle layer between the C++ application and any web application.

Client's decision to develop new framework

Once the decision to develop a java framework was finalized, the client approached ebusinessware with this assignment because ebusinessware already had a complete grasp of understanding of complex financial algorithm as well as the financial engineering expertise which included in-depth knowledge of C++, Java architecture and internal workings (JNI) of protocols between these two platforms.

Prior knowledge and experience of these technologies and domain alone saved weeks and weeks of learning curve and got the ebusinessware project team on the "get to work" track from day-one.

The step marked the beginning of a fruitful relationship between both the parties wherein two more projects were initiated after the first one in the same series. The subsequent project whose need arose during the last stages of the first one was to develop a generic test framework in JAVA as well as C++ to test the two systems discussed beforehand. The motive of this phase of work was to test both the systems for concurrency, thread safety, performance, stress and regression testing and accuracy of results. The last project in the ongoing trilogy was to cater to the following areas:

- To improve upon the latency, throughput, and response time of the JAVA Analytics System relative to the C++ Analytics System.
- To design JAVA wrappers for a set of utilities like LIBOR and Day Count Basis calculations.

To test the performance of the two systems for accuracy and variance in results across a spectrum of different hardware and software platforms, the platforms used were Windows XP/2000, LINUX x86 and SOLARIS SPARC.

The major challenges in the successive phases of the first project that ebusinessware successfully met:

The first phase work entailed exposing more than 270 native methods and bringing the functionality of those methods into Java layer in unenviable span of three and half months including the testing of all the methods and matching the results with the existing C++...



framework. The throughput and response time of the system was found to be comparable to that of AF coded in a native language.

The second phase work comprised of flexible testing infrastructure which enabled the developers to test the application for different run time parameters with easy approach of passing the parameters in XML and being able to store the results in a pre describable format.

In **The Third phase** work, the most demanding and arduous part of the project was porting and testing the system on multiple platforms. The two root parts of IRIS Analytics Framework were coded years back in non-standard C++ that are not supported by modern compilers and linkers.

Another intriguing part was to revamp the whole JAVA framework to minimize overhead incurred in cross language data exchange.

PROJECT OVERVIEW AND EBUSINESSWARE'S APPROACH

From the project requirements, it was clear to the Project Management team that:

- It was essential to bring together an offshore-based development team to achieve this. The objective was to:
 - Understand the client's decision drivers and requirements.
 - Analyze the architecture and source code (and obtain client sign-offs).
 - Develop and integrate the entire solution for the client's final review.
- The entire system needed to be modularized and documented for easy maintainability and extensibility.

Solution

The client approached ebusinessware to propose a solution to provide a JAVA wrapper for the existing application. A series of meetings followed between ebusinessware onshore unit and the client team. The project was thence passed on to ebusinessware offshore unit. All three phases were done with 100 percent offshore involvement.

The salient points of approach and achievements of all the projects are briefly mentioned in sequence as under:

First Phase

- The throughput and response time of the system is comparable to that of AF coded in a native language.
- The system provides a multithreaded interface to the in place Analytics Framework (AF). Java layer helped to bring out the business logic and calculations that is performed by the stand-alone application.
- By enabling Java layer, application can run on any platform and web server.
- The system can also support multithreaded concurrent access to single threaded AF.
- System uses intelligent caching mechanism to avoid expensive inter system calls

Second Phase

- A test suite in JAVA as well as C++ was developed to run a specific combination of tests
- A set of XML files for providing input and a set of output files plus a log file for output.
- The APIs can be tested in a sequential order or in random order.
- The framework can be used to test single as well as multithreaded system.
- Each test can be run single or multiple times from different threads.
- Erroneous or exceptional conditions can be logged and reproduced for comparison.
- Tests can be run against a number of entities by providing the list of entities and their attributes through another XML file.
- A single API can be run multiple times by providing a range of input values incremented in



Third Phase

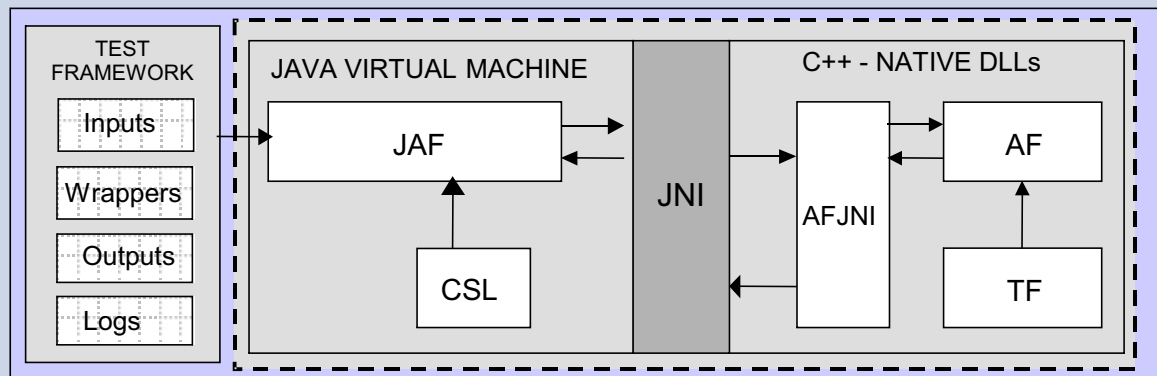
- The throughput of the system was improved by restructuring cross language calls to minimize overhead and improve response time of the JAVA/JNI framework. The method employed was so effective that the latency period was reduced to 1/4th in heavily used utilities.
- The JAVA and C++ frameworks were tested against each other for performance, variance in results etc. after applying enhancements to the JAVA framework.
- The state of the art test framework was ported across various UNIX based platforms upon x86 and SPARC. EBW insured that minimal changes had to be done to the underlying code to obtain platform independence.

The Analytics frameworks were rigorously tested to establish complete portability of the system across a spectrum of software and hardware platforms.

Technical Architecture

The diagram shows that Java Analytics Framework (JAF), which is the API wrapper and calls the AF via JNI providing Inputs and get the resultant. The test framework will call the JAF passing test arguments through XML files. The framework will provide the facility to write the log and output in different files.

ARCHITECTURE OVERVIEW



JAF – Java Analytics Framework
CSL– IRIS Foundation Classes

AFJNI – Analytics Framework JNI
AF – C++ Analytics Framework
TF – Technical Framework

Salient features of the architecture

- Providing a solid foundation on which enhancements can take place easily.
- The client's architecture allows Java clients to use the C++ analytic framework without having to make any changes to it enabling smooth integration with the client Application and Java clients.
- However, in order to provide clients with smoother and seamless transition, this elegant solution relies on a large number of small and medium sized steps making the overall architecture seem complicated. Once learnt and experienced with it, this architecture seems easy enough but to a new participant learning initial details and getting a grasp on the application can be daunting due to very large number of involved Model Calculations.

Timeframe

The client sought delivery within three months. Ebusinessware recommended making a priority list of deliverables. It was stipulated that In the worst-case scenario, the client would receive the top most items of the priority list within the required time span.



Documentation norms

It was also decided that a status report would be created for every week once the project started. This report would be discussed during a weekly status meeting/call with all stakeholders. It would provide the details of tasks accomplished in the week; plan for next week; quality assurance activities during the week and important open issues on the project.

TEAM ORGANIZATION

Division of responsibilities

The unique point about this project was that it was fully handled by the offshore unit of ebusinessware through direct client interaction. Only the first phase initiation took place between the onshore team and the client team. The timelines of the first project being very demanding, a team of nine developers (6 Java and 3 C++) and 2 testers was formed.

The second phase work of developing a test framework and testing consisted of a team of six developers and a tester and it also went to be a complete success in term of on schedule delivery and client satisfaction.

The team for last phase work comprised three full time developers one each for Java, C++ and testing.

Project Roles and Responsibilities

The Project Lead guided the team with regards to architecture, technology nuances, finance related questions, and ensured highest level of productivity, besides interacting with the onshore office and the client in terms of status updates and clarifications. The developers fixed the programs to comply with client recommendations and for error handling. The documenter ensured that all appropriate documentation from the AF ".C" and ".h" had been transferred to corresponding methods in the JAF programs. The Quality Analyst tested all java test programs. The infrastructure member ensured that existing IRIS application was installed and that existing C and Java test programs executed and produced outputs.

PROOF OF CONCEPT

It was stipulated that after the project had been accurately scoped, the Proof of Concept (POC) would start. During the POC stage, ebusinessware's thrust was on to lay foundation of the required and recommended framework and finish the POC successfully, so as to be able to carry on the with the rest of the development.

The team realized that any changes to POC at a later stage would distract the team from specified milestones which would imply changes to the work already having been done.

The POC milestone was broken into related (sub)milestones.

Foundation setting: It involved agreeing upon and implementing recommended Javadoc framework; Error handling framework; Enum changes to the existing framework to comply with the new requirement; Package naming convention changed to the existing framework to comply with new requirement and possibly, a different caching strategy as indicated by the client.

POC delivery: A round trip from the Java Test program > JAF > AFJNI > AF > AFJNI > JAF > Java Test Program. The ebsuisnessware team would take initial architecture and available code base from the client and implement it after ensuring that the client's recommendation had been incorporated.



CHALLENGE

The client's framework for one of their financial applications, handling Risk management calculation, Fixed Income, Futures and Option Calculation etc., was developed in C++ environment, which was a major hindrance as it could not be used from the remote location or from any web based application.

INNOVATION

ebusinessware created a JAVA interface, which acts as a middle layer between C++ application and the web application. With this innovation the throughput and response time of the system is comparable to that of AF coded in a native language; the system provides a multithreaded interface to the existing framework; Java layer helped to bring out the business logic and calculations performed by the stand-alone application. By enabling the Java layer, the application can run on any platform and web server. The system can also support multithreaded concurrent access to single threaded AF. The System uses intelligent caching mechanism to avoid expensive inter system calls.

CONCLUSION

Client Feedback The client team's point of contact expressed great satisfaction with the work to date and the working relationship. Currently, ebusinessware is working on several new projects for the same client. The CTO of the client company has mentioned, 'a very positive experience while working with ebusinessware team'.

The first two projects were a great success as they were delivered ahead of time. Minimum cosmetic changes were desired by the client after review. A very important aspect of the on-time delivery was the weekly status update conference between the ebusinessware offshore team and the client. The same methodology of communication was followed in later phases and the results were highly beneficial.

In the last phase, all development and porting work have been completed and the testing on SOLARIS SPARC platform is in progress.

our value proposition

