JPMorgan Raises the Bar for Banking Applications

JPMorgan

As one of the world's leading financial institutions, JPMorgan tracks changes in the world's financial markets 24 hours a day to bring their customers competitive retail, investment, commercial, and mergers and acquisitions banking services. With offices around the world and many highly specialized IT requirements for critical business functions such as inventory management and price forecasting, JPMorgan has built up a wide assortment of custom-built applications designed to meet specific needs.

Islands of Development

Such organic development, often at the departmental or branch level, is common in

"We wanted to develop an alternative to spreadsheets as a platform for modeling applications."

needs of its employees, it also has real drawbacks that become more significant as markets globalize and their IT environments become more complex.

For example, since many applications have been written to solve narrowly defined problems, it is often difficult to reuse their functionality to solve similar problems in new software development.

The speed with which an application can be deployed in the front office to support a new business opportunity can have a significant impact on the value of the new offering to

Snapper and One Bench provide an alternative to spreadsheets for real time grid based applications.

the financial services industry, and has served JPMorgan well for many years.

But while decentralized software development gives JPMorgan valuable creative flexibility to meet the specialized

the firm. Rapidly deployed, tactical systems often result in cases where the new product offering takes off, the scaling of these systems from low volume niche

applications, into industrial strength strategic platforms can be a real challenge.

A case in point is large number of pricing applications developed as Excel spreadsheets that perform and display price calculations on the trader's desktops, the applications shortcomings were becoming more apparent as the demands on them increased. As Bruce Skingle, Distinguished Engineer with the Investment Bank Technology group at JPMorgan relates, "Spreadsheet applications are quick to deploy and can be understood and modified on the trading floor. This can be a powerful tool for a novel product, but it opens up a host of control issues. With more mature products the need for consistent pricing and risk management across regions becomes more important." It was also difficult to back up data associated with these custom copies.

Other problems noted by Skingle included general performance and reliability issues.

Throughout the bank, other groups faced similar problems. For example, building financial applications to take advantage of new market conditions involved a cumbersome combination of Excel and C++; and took too long to build. Similarly, other groups faced the challenge of maintaining the many applications for entering reference data that had been developed in isolation over the years, duplicating functionality and code. Up to ten years old and written in C++, they did not share functionality and were burdensome to maintain.

"We needed to know that we could recover in a hurry if an update went wrong. Having applications down for any amount of time on the trading floor is not an option."

Bridging Developers

It was against this backdrop that the Investment Bank Technology group started work on a replacement for their existing systems. More than a re-write of the existing spreadsheets, they wanted to lay a solid foundation for enhanced security, auditing, scalability, interoperability and, above all, reusability. To achieve this, Skingle understood that they needed to produce a generalized platform in which this and other new applications could be extended, reused and shared. As he explains, "We wanted to develop an alternative to spreadsheets as a platform for modeling applications and at the same time allow for greater abstraction and reuse across other projects. As a tool for building a new model, a spreadsheet is an excellent tool, but by the time a product is traded the official models developed by the Quantitive Research group are delivered as libraries, and the spreadsheet is being used as little more than a .dll container and a grid based GUI." To this end Skingle envisioned that the spreadsheet-based GUI and code for common tasks like single sign-on and logging would be openly available for developers using the new environment, leaving application developers free to focus on the business problem.

The result was One Bench, a platform for developing and delivering custom banking applications, starting with Fuse, their new bond trading application.

Fuse is a perfect example of a rich client banking application. Receiving a constant

stream of market data on bond trading prices, the application consolidates disparate sources to traders in a familiar grid. This data can then be used to help traders identify trends and perform 'what-if' analysis. The designers knew that, as a rich client application, Fuse consumes data in streams that could be shared across other bank applications. They envisioned a GUI framework, called Snapper, as another shared component that would run in the One Bench environment and into which developers could snap applications to provide a consistent user view to similar data.

EIS is being developed on One Bench by the IB Technology Exotics & Hybrids group to consolidate the large number of systems that have been set up over the years for entering reference data. Up to ten years old and written in C++, they did not share functionality and were burdensome to maintain. As Martin Game, Vice President with the IB Technology Exotics & Hybrids group relates "Moving our first project to Eclipse and One Bench took several months, as we added shared functionality to the environment, but future work will benefit from that investment and make us far more efficient." Mr. Game expects that they will have several more EIS reference data applications ported to One Bench by the end of the year.

Similarly, QTrade was developed independently by the Exotics & Hybrids group in the bank to solve the problem of difficult and slow to write financial applications. Banks produce new applications on very short notice to take advantage of financial trends. Previously, each new application involved a combination of Excel and customized C++ coding, with a turn-around time of over two weeks.

JPMorgan evaluated several options for the development of One Bench, but the final choice came down to Java/Swing or Eclipse RCP. After careful evaluation, RCP emerged as the better alternative. RCP is designed from the ground up as a plug-in platform that will allow One Bench applications to be written as Eclipse plug-ins.

Hitting the Ground Running

Support for native windowing was also a significant advantage. Applications built for One Bench will meet users' expectations for familiarity. Taking advantage of One Bench and Eclipse, the QTrade developers were able to develop a flexible and easy-to-use UI based on SWT and XML. Users can structure new exotic financial products using simple drag-and-drop operations to add libraries to a canvas and add business logic, and development times are reduced from weeks to hours.

Graphical performance was another area where RCP excelled. Previous development of Swing based applications had involved significant efforts to get performance from the very functional table control with large data sets and rapid update rates. A Snapper based proof of concept has demonstrated that an SWT-based table is capable of maintaining a 400,000 row data set with 1000 updates per second without any special coding tricks or unusual hardware. The IB Technology Exotics & Hybrids group's EIS project saw similar benefits from RCP's performance.

Delivering Mission Critical Applications

Since the long-term plan is that One Bench will become the platform of choice for developers of desktop applications within JPMorgan, the logistics of maintenance were critical. For instance, not all users need or are entitled to the same applications, or

specific functionality within an application. Roll out and updates had to be automated based on a user's credentials. Equally important, when deploying multiple applications to the same Java Virtual Machine, there is always the risk that a problem with updates to one application would crash one or more critical applications. "We needed to know that we could recover in a hurry if an update went wrong. Having applications down for any amount of time on the trading floor is not an option."

JPMorgan's Single Sign On system was originally designed for web-based applications and they can integrate this system with One Bench so that the system knows exactly what updates to apply to each desktop. Advanced features such as the ability to roll an application back to a previous release state, are also critical. Skingle explains that "banks are very conservative about any new technology, and sharing applications on a JVM is a real concern. Automated roll-backs in the

Model Outputs	Basket Average		CashSettleDate	IR Definition
This form contains model outputs	Basket Average Parameters		Cash Settlement Date	IR Definition Parameters
Price	Strike	1.1		Currency USD
	Is Call	true	Date 26-MAY-2006	
	Weight Type	U	EOD EOD	
	Smile Type	D		
	Ref=Avg Out ?	N		
	Notional	1		
	Value Date(Date)	26-MAY-2006		
	Value Date(Time)	SOD		
	Instrument Settlement	CashSettleDate		
	Ignore Past Values Dates ☐			

Equity Basket

Equity Basket Parameters

Currency USD

Asset	Weight	Style

JPMorgan used SWT and XML to develop the easy-to-use QTrade application authoring tool. Users drag components to a canvas and apply business logic to develop applications quickly and without programming.

option."

These requirements made Eclipse's Update Manager central to their strategy for One Bench and the components and applications such as Snapper, EIS and QTrade that run on it. Accessing the Update Manager's API directly, they can take complete programmatic control of the update process. For example, they by-pass end user input, performing all application updates silently.

Eclipse Update Manager reduce our risk and makes the open One Bench approach feasible." QTrade applications, for example, are saved as XML files, and these are easily deployed out to desktops using the Eclipse Update Manager. As Paul Sampat, Vice President with the IB Technology Exotics & Hybrids group explains, "The One Bench plus Update Manager combination lets us develop and deploy quickly while reducing our risk."

Indeed, with the Update Manager, an application can be updated globally over a weekend. If a problem is discovered when employees start work on the following Monday in the Far East, the situation can be corrected immediately, and the rest of the global workforce will never know there was an issue.

Eclipse RCP gives JPMorgan what they need: a plug-in friendly environment that will encourage efficiency and lower costs, and the control to ensure that software can be managed effectively.

All of these development groups had independently identified Eclipse as their environment of choice. When they learned of the One Bench initiative, they each knew that it would make their tasks easier and make sense for JPMorgan. One Bench gives them a head start for their own requirements, and as they contribute components, they make it a more attractive solution for developers throughout the bank.

With One Bench in place, they expect their developers to develop the banking world's next killer-applications.

Ron Stone is a technology writer and content management consultant based in Ottawa, Canada

![eclipse logo]

http://www.fbcoverup.com/docs/eclipse/2007-03-06-JPMorgan-Eclipse-
Foundation-Community-Awards-Announcement-ECLPISECON-2007-
Mar-06-2007.pdf

☐ DOWNLOAD

HOME / ABOUT US / ECLIPSE FOUNDATION COMMUNITY AWARDS...

About Us

☐ Foundation

☐ Governance

☐ Legal Resources

☐ Contact Us

Eclipse Foundation Community Awards Announcement

Press Release

Eclipse Community Awards Winners Announced

Awards Recognize Top Individuals and Technologies in the Eclipse Community

ECLIPSECON 2007, SANTA CLARA, Calif., March 06, 2007�The Eclipse Foundation today announced the
winners of its annual Eclipse Community Awards, which recognize the top individuals, products and organizations

in the Eclipse community. Recipients were honored at an awards ceremony yesterday on the opening day of EclipseCon 2007.

The Individual Awards acknowledge people that have given time and resources to make Eclipse a better community. Winners were selected by their peers in an online vote. The selection of the �Top Newcomer Evangelist� award was based on the individual that answered the most questions raised on the eclipse.newcomer newsgroup.

- **Top Ambassador**: Chris Aniszczyk. Finalists in this category are Alex Blewitt and Doug Schaefer.
- **Top Contributor**: Kimberley Peter and Tom Schindl (tie). Finalists in this category are Eugene Kuleshov and Philippe Ombredanne.
- **Top Committer**: Ed Merks. Finalists in this category are Mik Kersten and Thomas Watson.
- **Top Newcomer Evangelist**: Daniel Megert. Finalists in this category are Hendrik Maryns and Eric Rizzo.

The Technology Awards recognize both Eclipse-based open source projects and commercial products. Winners were selected by a panel of judges from the Eclipse community. Nominated RCP applications and developer tools were evaluated on their user interfaces, innovation and comprehensiveness. Organizations competing for the �Best Deployment of Eclipse Technology in an Enterprise Technology� award were evaluated based on number of Eclipse projects being used, scale of applications being deployed and the scope of deployment.

- **Best Open Source RCP application**: PSICAT (**http://portal.chronos.org/psicat-site**). The finalists are EclipseTrader (**http://eclipsetrader.sourceforge.net**) and RSSOwl (**http://www.rssowl.org**).
- **Best Commercial RCP application**: TIBCO Business Studio (**http://www.tibco.com/devnet/business_studio**). The finalist is Ivis Technologies xProcess (**http://www.ivis.com/public/products/xprocess**).
- **Best Open Source Eclipse-based Developer tool**: eclipse-cs Checkstyle Plugin (**http://eclipse-cs.sourceforge.net**).

    The finalist is EclEmma (**http://www.eclemma.org**).
- **Best Commercial Eclipse-based Developer tool**: QNX Momentics IDE (**http://www.qnx.com/products/development**)). The finalist is Adobe Flex 2.0 (**http://www.adobe.com/products/flex**).
- **Best Deployment of Eclipse technology in an enterprise**: JPMorgan Chase (**http://www.jpmchase.com**).

About the Eclipse Foundation
Eclipse is an open source community whose projects are focused on providing an extensible development platform and application frameworks for building software. Eclipse provides extensible tools and frameworks that span the software development lifecycle, including support for modeling, language development environments for Java, C/C++ and others, testing and performance, business intelligence, rich client applications and embedded development. A large, vibrant ecosystem of major technology vendors, innovative start-ups, universities and research institutions and individuals extend, complement and support the Eclipse Platform.

The Eclipse Foundation is a not-for-profit, member supported corporation that hosts the Eclipse projects. Full details of Eclipse and the Eclipse Foundation are available at **www.eclipse.org**.

# # #

All company/product names and service marks may be trademarks or registered trademarks of their respective companies.

Press Contact
John Moran or Lori Kroll
Schwartz Communications
781-684-0770
eclipse@schwartz-pr.com

Eclipse Foundation

About us

Contact Us

Donate

Governance

Logo and Artwork

Board of Directors

Legal

Privacy Policy

Terms of Use

Copyright Agent

Eclipse Public License

Legal Resources

Useful Links

Report a Bug

Documentation

How to Contribute

Mailing Lists

Forums

Marketplace

Other

IDE and Tools

Community of Projects

Working Groups