ENABLING THE FUTURE

Natural / Adabas

Migration Solutions

Why EGL?

1 RBD/EGL - the best Application Development Environment for migrating from Natural!

The IBM and FBDA primary recommendations for choice of an Application Development Environment and programming language when migrating from Natural are Rational Business Developer (RBD) and its associated 4GL programming language, EGL.

We recognize that with the buzz about Java and the support of the open source community this recommendation may seem at odds with perceived industry trends.

This paper provides a strong rationale for the recommendations from a number of perspectives.

2 From a Strategic Business Perspective

From a strategic business perspective:

- RBD and EGL are part of the IBM Rational strategic plan. They are technologies for the future, and are supported by significant investment, marketing and sales programs.

- As part of a strategic legacy modernization initiative within the IBM Rational brand, RBD and EGL have been selected for ***all*** IBM Rational Application Modernization efforts. IBM Rational has developed, and will continue to develop strategic relationships with multiple migration partners such as FBDA to support this initiative.

- IBM modernization partners have agreed to support the transformation of legacy application technologies to EGL. IBM Rational recognizes the importance of customer demands for cost effective, high quality legacy modernization capabilities and will continue to support these important customer needs with technology advances in EGL and partner modernization capabilities.

3 From a Technology Perspective

From a technology perspective:

- RBD and EGL help IT organizations build innovative solutions while controlling costs, reducing application backlogs and improving flexibility and responsiveness to the business.

- Using RBD, Customers can achieve increased levels of application development productivity and enable business-savvy developers to exploit emerging computing technologies to fulfill new and changing business requirements with minimal retraining costs and time.

- RBD and EGL will allow Customer to easily reuse and leverage valuable existing IT assets; particularly when migrating from proven Natural applications with typically robust business functionality.

- RBD is the component of the IBM Rational Software Delivery Platform that's specifically designed to fulfill the needs of business-oriented developers

- RBD provides a comprehensive development workbench for the IBM Rational EGL language, a powerful, easy-to learn and highly productive modern language that equips developers of almost any background with a simplified and more abstract development paradigm that allows them to quickly deliver cross-platform, transactional data-centric services and applications.

4 From a Business Programming Language Perspective

From a business programming language perspective:

- EGL is designed to let you write full-function applications quickly and independently from the target run-time platform, freeing you to focus on business problems rather than complex software technologies. For example, it hides the Java™ and Java Platform, Enterprise Edition (J2EE) details.

- EGL hides the Web services standards and the enabling middleware so you can deliver enterprise data to employees, customers and partners through Web browsers with minimal Web technology experience.

- EGL hides the details of the target execution platforms and associated middleware, allowing the application developer to focus exclusively on the business problem rather than on the underlying implementation technologies.

- Developers with Natural experience but who have little or no experience with Java and Web technologies can readily preserve and leverage their procedural programming skills and valuable business domain expertise via EGL to create enterprise-class services and applications quickly and easily.

- the RBD/EGL combination provides state-of-the-art tools that allow developers direct access to a broad range of EGL construction, test/debug, and deployment tools, helping to enable IT organizations to create Web, Web service, batch character-based and graphical UI applications quickly and easily.

- Abstraction. EGL provides concise and powerful notations that eliminate the tight coupling and reduce the amount of coding required to interface systems and middleware. This abstraction can significantly simplify and speed up developers' work.

- Declarative programming. EGL includes, where possible and appropriate, a certain level of declarative specifications aimed at reducing repetitive and error-prone coding. For example, by associating a validation rule to a data item, every time the item is used in a certain context, the validation is automatically applied and enforced.

- Language. EGL provides a comprehensive but easy-to-learn language notation that's modern and modular. And it includes a rich library of built-in functions to boost your productivity for commonly required operations, such as date and time math, string manipulation and more.

- Additionally, the language is extensible and offers full interoperability with other languages; in particular, it provides EGL interfaces to native Java code.

- the RBD/EGL combination includes a runtime code generation engine that transforms the EGL source into Java or COBOL code optimized for deployment to a broad variety of application hosting environments including J2EE servers (WebSphere, Apache Tomcat, etc.) and traditional transactional systems such as System z Customer Information Control System (CICS) or System i5OS.

5 From a Business Application Developer Perspective

The RBD/EGL combination empowers a broad class of developers with either a background in the existing Natural applications or with newly developed skills to:

- quickly create highly optimized services and applications, enabling the Customer to create a flexible pool of development resources that are portable and adaptable across projects and platforms.
- build Java/J2EE/JSF code-based applications without having a deep knowledge of the underlying Java technologies.
- deliver applications based on industry standards that interoperate with existing systems.
- adopt a service orientation application development paradigm without extensive knowledge of SOA supporting technologies and standards.
- adopt new technologies more efficiently with lower training and learning curve costs.
- achieve the highest level of productivity while leveraging the latest platforms and technologies.
- reduce application errors through abstraction, code generation and automation.
- deploy to all IBM platforms and multiple non-IBM platforms.

6 From a Systems Integration Perspective

Regardless of the developer's background, you can use IBM RBD and EGL to quickly generate applications and services that will deploy natively to the broadest variety of platforms. EGL is uniquely designed to address a full spectrum of business application requirements by allowing development of:

- Business services. The language includes the built-in notion of "service," allowing you to create and consume services in an extremely simple and straightforward way, and to permeate systems architecture with service orientation.
- Web applications. Tight integration of EGL with the JavaServerFaces (JSF) framework and JSF tooling enables you to create Web applications in a simple and productive way, without your needing to know Java or the details of the JSF framework.
- Reports. EGL integration with the Jasper Reports, an open-source reporting engine, allows you to create professional reports.

- Batch systems. The language includes the built-in notion of "batch program," which can be generated to run without end-user interaction, for example, for reports production or for batch database load/update.

- GUI applications. In addition to the Java Visual Editor application, which supports development of Standard Widget Technology (SWT) and SWING clients, RBD also includes EGL support for defining and running a simplified GUI client mode.

7 Summary

IBM Rational Business Developer Extension empowers business-oriented developers to be extremely productive quickly as a result of the following capabilities:

- Programming Language. A powerful easy to learn programming language with a wide variety of features and capabilities to simplify application development, debug and deployment to multiple platforms

- Generation. Although simplified, the EGL development technology must still guarantee optimal deployment to the run-time platforms to take advantage of their qualities of service and to allow native management and monitoring of the systems in operation. This is accomplished through a code generation engine included within RBD that transforms the EGL source code into native Java or COBOL source, and creates any other required deployment artifacts.

- Tools. To further boost developers' productivity, Rational Business Developer Extension includes a rich set of tools built upon the **Eclipse** integrated development environment (IDE) framework. These include EGL source animation for debugging; powerful smart editing; visual construction; graphical navigation; and specialized capabilities such as tight integration of EGL notations with graphical Web development tooling and automatic transformation of Unified Modeling Language (UML) models or database schemas into completely functional EGL services and applications.

In short, we are recommend the RBD/EGL combination because we believe strongly it is the best strategic technology choice for customers with demanding application development and maintenance requirements while also providing a cost effective migration path for Natural users applications and development staff.