

RPG Anwendungen modernisieren mit IBMs Enterprise Generation Language

Mit der Enterprise Generation Language EGL eröffnet die System i Developer Roadmap interessante Perspektiven für einfache und moderne Anwendungsentwicklung auf System i. EGL unterstützt von der klassischen Anwendungsprogrammierung, über die Erstellung von Web Anwendungen bis hin zur Serviceorientierten Programmierung alle wesentlichen Anforderungen an moderne Programmierung in einem Tool und einer Sprache. Durch die einfache Erlernbarkeit haben auch RPG Programmierer nun die Möglichkeit ohne viel komplizierte Technik Web Anwendungen zu schreiben und ihre vorhandenen Anwendungen Schritt für Schritt in eine Service- und Geschäftsprozessorientierte Architektur zu überführen.

EGL hat das Potential Anwendungsentwicklung für System i wieder attraktiv zu machen. Gab es bisher für RPG Entwickler wenig Perspektiven außer einem radikalen Wechsel zu .NET oder Java, gibt es mit EGL die Chance im Rahmen des Tagesgeschäfts sukzessive auf moderne Programmierung umzusteigen.

In den letzten Jahren haben viele Unternehmen sich damit befasst graphische Benutzeroberflächen für die RPG Anwendungen zu erstellen. Basierend auf dem 5250 Datenstrom konnte mit Werkzeugen wie z.B. Webfacing oder HATS die Optik der Anwendungen teilweise deutlich verbessert werden. Was die Erzeugung echter funktionaler Mehrwerte betrifft, hatte dieses Screenscraping genannte Verfahren allerdings enge Grenzen. Wirklich attraktive neue Funktionalität lässt sich in Kombination von RPG, 5250 Datenstrom und Screenscraper nur mühsam entwickeln. Mit EGL, seinem komfortablen graphischen Screen Designer und seiner serviceorientierten Programmarchitektur lässt sich neue Funktionalität erheblich einfacher und wartungsfreundlicher entwickeln. Optimal ist eine sinnvolle Kombination aus neuer EGL Funktionalität und bestehender RPG Logik.

Um nun für die Anwender eine möglichst einheitliche und attraktive Benutzeroberfläche zu schaffen die leicht mit neuer Funktionalität ergänzt werden kann, hat PKS mit dem Server Builder 400 eine Lösung geschaffen die eine nahtlose Integration von RPG und EGL Benutzeroberfläche ermöglicht. So kann sukzessive Funktionalität erneuert werden ohne dass der Anwender einen Unterschied zwischen RPG und EGL bemerkt.

Im ersten Schritt wird hierfür die bestehende RPG Anwendung mit Server Builder Serverfähig gemacht und mit einer graphischen Oberfläche (Windows oder Web) versehen. Neue EGL Funktionalität lässt sich dann einfach und direkt in die bestehende Anwendung einbinden. So lässt sich die Anwendung Schritt für Schritt modernisieren ohne dass der Anwender einen Technologiebruch bemerkt.

Mit Unterstützung des EGL Entwicklungslabors hat PKS darüber hinaus ein Verfahren entwickelt mit dem bestehende RPG Programme sogar nach EGL übertragen werden können. Bestehende Funktionalität lässt sich somit ohne aufwendige Neuentwicklung in EGL weiterentwickeln. Dieser Übergang kann in kleinen Schritten erfolgen die sich leicht ins Tagesgeschäft einbauen lassen und somit die Anwendung sukzessive modernisieren.

Wenn die Programme auf EGL umgestellt sind, dann kann EGL entweder schnellen native Code für System i oder sogar plattformunabhängigen Java Code erzeugen. Für RPG Entwickler ist dies ein sehr effizienter Weg um auch Software für Windows, Linux, Unix oder sogar Mainframes schreiben zu können. Der vorhandene Code und das vorhandene Know How werden somit plattformunabhängig.

Für die Modernisierung sind folgende Schritte sinnvoll:

1. Umstellung der RPG Entwicklung auf WDSC. Die graphische Entwicklungsumgebung ist der erste Schritt zur modernen Programmierung.

2. Serverisierung der RPG Anwendung mit Server Builder und erstellen einer graphischen Benutzeroberfläche
3. Erweiterung der WDSC Entwicklungsumgebung um RBDe. RBDe enthält EGL und eine Reihe interessanter Werkzeuge zur Entwicklung von Geschäftsanwendungen wie z.B. graphische Datenmodellierung von DB2 Datenbanken, Erstellung von Web Services etc. Über unterschiedliche Sichten kann leicht zwischen RPG Programmierung, EGL Programmierung und Datenbankmodellierung gewechselt werden
4. Erstellung neuer Funktionalität in EGL für Anwendungserweiterung, Verbesserung von Bedienabläufen, Strukturierung von Funktionalität in Geschäftsprozessen, Integration mit Fremdanwendungen, verbessertes Reporting, etc.
5. Übertragung von vorhandenen RPG Programmen mit den Migration Tools 400 EGL nach EGL um die neuen Möglichkeiten auch für die bestehenden Programme verfügbar zu machen
6. Optional Ersetzen plattformspezifischer Funktionalität falls eine komplette Plattformunabhängigkeit gewünscht wird. Dies kann manuell erfolgen oder durch Verwendung der Service Library 400, die System i spezifische Funktionalität plattformunabhängig zur Verfügung stellt

Um sich mit dem Thema EGL näher zu befassen gibt es eine Reihe von Veranstaltungen von IBM und PKS sowie Vorträge und ein Labor auf der iSociety World (ehemals Common Jahreskonferenz). PKS-Events unter www.pks.de/events

The image shows a screenshot of the IBM Rational Software Development Platform (RSDP) interface. The main window displays a web application titled "EGL Pet Shop demo" with a search bar and product listings. Several callout boxes provide detailed information about the development environment and its capabilities:

- Project Explorer:** Application Artifacts include Records, Data Items, Page Handlers, Libraries, Pages, Styles, Templates, etc...
- Page Data:** Drag and Drop EGL Data Model Records and Data Items to build
- Control Attributes:** Customize visual, formatting, validation, paging
- Page Designer:** JavaServer Faces based GUI Page
- Command Event:** Trigger Server side EGL
- Control Palette:** Faces Components, Faces Client Components, HTML
- Business Logic:** Interactive logic development and debugging in EGL (For developers experienced in COBOL, RPG, PL/SQL, PowerBuilder, Informix, Visual