

**Assetmodernisierung mit IBM Rational Business Developer und EGL**

**White Paper**

Juli 2008

**Rational** software



## **Mehr Wertschöpfung aus RPG-Anwendungen auf der IBM i Plattform**

---

<b>Inhalt</b>	
<b>2</b>	<b><i>Einführung</i></b>
<b>3</b>	<b><i>Wann ist die Modernisierung von RPG-Anwendungen sinnvoll?</i></b>
<b>6</b>	<b><i>Schutz von IT-Investitionen</i></b>
<b>6</b>	<b><i>Gründe für die Migration auf IBM Rational Business Developer</i></b>
<b>8</b>	<b><i>Szenarios zur Anwendungs-transformation</i></b>
<b>11</b>	<b><i>Fazit</i></b>
<b>12</b>	<b><i>Weitere Informationen</i></b>

## **Einführung**

Anwendungen, die auf der Programmiersprache RPG aufbauen und unter dem Betriebssystem IBM i auf der IBM Power Systems-Plattform laufen, haben in den Geschäftsprozessen vieler Unternehmen eine zentrale Bedeutung. Während die ersten RPG-Anwendungen immer noch so ausgeführt werden wie früher, wurden spätere Versionen mit der Zeit immer umfassender und komplexer. Ein Markenzeichen des Betriebssystems IBM i war schon immer, Upgrades auf neue Releases vornehmen zu können, ohne Anwendungsprogramme ändern zu müssen. Das Dilemma war jedoch, dass die Funktionen, die für Stabilität in der Kundenumgebung sorgten, auch mehr Komplexität ins Unternehmen brachten, und zwar durch Technologien, die im Verlauf der vergangenen 15 bis 20 Jahre jeweils den aktuellen Stand darstellten. Durch die Sprachenkompatibilität war der Kunde zudem in der Lage, im Lauf der Zeit stufenweise neue RPG-Sprachenfunktionen zu implementieren. Daraus resultierte letztendlich eine Mischung aus aktuellen und veralteten Sprachenkomponenten und -funktionen.

RPG und 5250-Anzeigen bildeten bisher die Grundlage der IBM i basierten Kundenumgebungen. Mittlerweile hat sich dies jedoch grundlegend geändert und es hat ein Wechsel zu neuen Benutzerschnittstellen wie Java™- und Microsoft® .NET-Umgebungen stattgefunden. Diese Umgebungen haben sich sehr schnell als Standard etabliert und stellen praktisch einen Generationswechsel in der Anwendungsentwicklung dar. Um auch in Zukunft wettbewerbsfähig zu bleiben und moderne Architekturen und Middlewarelösungen nutzen zu können, müssen sich die Unternehmen einem grundlegenden Wandel unterziehen. Dies belegt ein Auszug aus einer kürzlich von Gartner veröffentlichten Pressemitteilung: „Obwohl die IT-Management-Teams durchaus in der Lage sein werden, die Systeme noch eine gewisse Zeit zu unterstützen, werden Geschäftsbenutzer in absehbarer Zeit nicht mehr bereit sein, mit Lösungen, die keine modernen Standards in Bezug auf Funktionalität und Flexibilität unterstützen, weiterhin zu arbeiten<sup>1</sup>.

<sup>1</sup> Gartner, „Gartner Says IT Leaders Must Place IT Modernization at the Core of Their 2008 Objectives“, Pressemitteilung, 26. Februar 2008, [www.gartner.com/it/page.jsp?id=611507](http://www.gartner.com/it/page.jsp?id=611507).

---

## Highlights

---

*Unternehmen können ihre vorhandenen IT-Assets wiederverwenden und somit flexibler reagieren, die Produktivität bei der Anwendungsentwicklung verbessern und neue Geschäftsmodelle unterstützen.*

*Hohe Wartungskosten, Rückgang bei der Anwendungsflexibilität oder Leistungsverschlechterungen bei der Entwicklungsplattform sind Anzeichen dafür, dass die Migration einer Anwendung auf eine neue Plattform notwendig ist.*

Die Unternehmen haben in der Vergangenheit umfassend in ihre traditionellen Systeme, Anwendungen und Mitarbeiter investiert. Für viele Unternehmen sind die vorhandenen Systeme und der RPG-Code nach wie vor Assets mit zahlreichen geschäftlichen Vorteilen. Damit dieser Generationswechsel aus wirtschaftlicher Sicht machbar ist, müssen die Unternehmen einen Großteil der bestehenden Anwendungen und des vorhandenen Know-hows auf die neue Entwicklungsumgebung umstellen, um eine Wiederverwendung zu ermöglichen. Gleichzeitig bilden sie den Ausgangspunkt für die schrittweise Modernisierung. Die Migration von RPG-Anwendungen stellt eine Option zur Realisierung dieses Transformationsansatzes dar.

Dieses White Paper zeigt auf, wie Unternehmen IBM Rational Business Developer einsetzen, um kosteneffizient mehr Flexibilität zu schaffen, die Produktivität bei der Anwendungsentwicklung zu optimieren und neue Geschäftsmodelle wie eine serviceorientierte Architektur (SOA) zu unterstützen – und dabei die vorhandenen IT-Assets weiter nutzen.

### **Wann ist die Modernisierung von RPG-Anwendungen sinnvoll?**

Viele Unternehmen investieren Unsummen in die Entwicklung individuell angepasster Anwendungen, um sich von Mitbewerbern abzuheben und ihre internen Geschäftsprozesse optimal zu unterstützen. Und trotzdem sind solche Anwendungen in Bezug auf Funktionalität und Technologie sehr schnell veraltet und verlieren ihre geschäftlichen Vorteile für das Unternehmen.

Durch kontinuierliche Wartung kann der Prozess der funktionalen Alterung etwas verlangsamt werden. Die technologische Alterung wird häufig jedoch erst dann festgestellt, wenn im Verlauf des Lebenszyklus unverhältnismäßig hohe Wartungskosten auflaufen, die Anwendungsflexibilität nachlässt oder die Entwicklungsplattform langsam aber sicher veraltet. Dies sind Anzeichen dafür, dass Anwendungen auf eine neue Technologieplattform migriert werden sollten, bevor die Kosten schwerer wiegen als die Vorteile.

---

## Highlights

---

*Verschiedene Faktoren machen eine Transformation erforderlich: Verlängerung des Anwendungslebenszyklus, Verfügbarkeit von Anwendungen auf anderen Plattformen, Wechsel zu einer SOA, effizientere GUI-Nutzung, bessere Beweglichkeit der Geschäftsabläufe, bessere Wettbewerbsfähigkeit, Rückgang des RPG-Entwickler-Know-hows und Konsolidierung der Entwicklerteams.*

Wenn dieser Punkt erreicht ist, befinden sich viele Unternehmen in einer schwierigen Lage. Sie stellen fest, dass es nicht besonders kosteneffizient ist, einfach nur die traditionellen Anwendungen durch moderne Standardsoftware zu ersetzen. Nur wenige Unternehmen können es sich erlauben, ihre angepasste Software auch weiterhin komplett umzuschreiben, um sie auf dem aktuellen Stand zu halten. Aber selbst diese Unternehmen erkennen mittlerweile, dass sie dieses Geld besser für einen kosteneffizienteren Ansatz zur Modernisierung und Migration auf aktuelle technische Standards ausgeben sollten. Zudem machen verschiedene andere Faktoren diesen Transformationsprozess praktisch unumgänglich.

- **Verlängerung des Lebenszyklus von Anwendungen.** Die *Wartung von älteren geschäftskritischen Anwendungen ist zunehmend schwieriger zu rechtfertigen, da immer weniger Geld für RPG-Schulungen bereitgestellt wird.*
- **Anwendungen müssen auf anderen Plattformen laufen.** *Im Rahmen neuer Plattformstrategien oder Konsolidierungsmaßnahmen müssen Anwendungen auch in anderen Umgebungen wie Linux<sup>®</sup>, Microsoft Windows<sup>®</sup>, UNIX<sup>®</sup> oder IBM System z laufen. Oder eine IBM DB2 Universal Database, ein Oracle- oder SQL-Server wird anstelle eines DB2-Servers in der IBM i Umgebung benötigt.*
- **Übergang zu einer SOA.** *Wenn bestehende monolithische Architekturen durch Geschäftsprozesse oder eine SOA aufgebrochen werden, ist die Unterstützung der einzelnen Assets über die Sprachen- und Entwicklungsumgebung von ausschlaggebender Bedeutung. Hinzu kommt, dass die optimale Unterstützung von Web-Services und Sprachen für die Modellierung von Geschäftsprozessen wie Business Process Execution Language (BPEL) gewährleistet sein muss.*
- **Effizientere Nutzung von GUIs.** *Benutzer, die in erster Linie mit Windows- oder webbasierten Anwendungen arbeiten, sehen den Wechsel zu Green-Screen-Anwendungen eher als eine Beeinträchtigung ihrer geschäftlichen Abläufe. Die Ursache liegt darin, dass es zwischen den Schnittstellen keine angemessenen Integrationsmöglichkeiten gibt. Zudem ist der erforderliche Schulungsaufwand sehr hoch. Auf Managementebene kann zudem der Eindruck erweckt werden, dass Green-Screen-Anwendungen als Indikator dafür gewertet werden, dass die IT-Abteilung nicht innovativ genug arbeitet.*

---

## Highlights

---

- **Bessere Beweglichkeit der Geschäftsabläufe.** Änderungen an Anwendungen, die sich immer weiter entwickeln, gestalten sich schwierig. Folglich kann sich die Einführung neuer Geschäftsprozesse sehr schnell kostspielig und zeitaufwendig gestalten und die Wettbewerbsfähigkeit des Unternehmens ernsthaft gefährden.
- **Bessere Wettbewerbsfähigkeit.** Da sich Java- oder Microsoft .NET-Anwendungen mittlerweile als neue Plattformstandards durchgesetzt haben, wird es für Anbieter von RPG-Standardsoftware immer problematischer, Kunden für ihre Produkte zu interessieren. Die Unternehmen müssen über den RPG-Tellerrand hinausblicken, um auch in Zukunft wettbewerbsfähig zu bleiben.
- **Zurückgehendes RPG-Entwickler-Know-how.** Neben dem fehlenden Innovationsansatz gestalten sich die Wartung und Entwicklung von bestehenden Anwendungen immer schwieriger, da immer weniger RPG-Entwickler-Know-how zur Verfügung steht.
- **Konsolidierung von Entwicklerteams.** In vielen Unternehmen ist in den Entwicklerteams eine Mischung aus RPG-, Java- und älteren 4GL- und COBOL-Anwendungen zu finden. Der Entwicklungsprozess wird dadurch unflexibler und teurer. Durch die Konsolidierung verschiedener Entwicklerteams können Unternehmen moderne Standardsprachen und Entwicklungsumgebungen nutzen, um die Prozesse flexibler und kostengünstiger zu gestalten.

**Transformationsinitiativen bilden die Basis für die Lösung dieser Problemstellungen. Die infrage kommenden Anwendungen müssen ein Minimum an Qualitätsstandards aufweisen, damit eine Transformation aus wirtschaftlicher Sicht sinnvoll ist.**

Neben Umstrukturierungs- und Modularisierungsmaßnahmen bilden Transformationsinitiativen die Basis für die Lösung dieser Problemstellungen. Laut Phil Murphy von Forrester Research „tendiert die IT dazu, bisher genutzte Technologien eher aufzugeben als wiederzuverwenden. Resultierend daraus wurden hierfür aus rein technologischen Gesichtspunkten Unsummen für den Technologeaustausch investiert, häufig auf Kosten des geschäftlichen Nutzens. Wertschöpfung für das Unternehmen ist das Schlagwort der IT des 21sten Jahrhunderts. Seitdem sind die CIOs damit beschäftigt, Konzepte aus der Vergangenheit mit aktuellen Anforderungen abzugleichen, woraus letztendlich das Interesse an der Modernisierung traditioneller Systeme und Anwendungen wiederbelebt wurde“<sup>2</sup>. Damit eine Transformation auch aus wirtschaftlicher Sicht sinnvoll ist, müssen die infrage kommenden Anwendungen ein Minimum an Qualitätsstandards aufweisen.

---

## Highlights

---

*Es gibt drei bewährte Ansätze für den Wandel bei der Anwendungsentwicklung: Austausch der Anwendung durch Standardsoftware, Neuschreiben des Codes und Migration auf eine neue Umgebung.*

### Schutz von IT-Investitionen

Es gab schon immer drei Ansätze, wenn es darum ging, einen allgemeinen Wandel bei der Anwendungsentwicklung herbeizuführen:

- **Austausch der Anwendungen durch Standardsoftware.** Der Austausch einer bestehenden Anwendung durch Standardsoftware eines Anbieters ist recht aufwendig, da dies auch organisatorische Änderungen und neue Abhängigkeiten von einem Anbieter nach sich zieht. Dies mag zwar sinnvoll für Standardprozesse sein, jedoch nicht für wettbewerbskritische Prozesse, bei denen das Differenzierungspotenzial sehr begrenzt ist.
- **Neuschreiben von Code. Das Neuschreiben von Code geht Hand in Hand mit hohen Kosten und Risiken bei der Zeitplanung.** Hinzu kommt, dass bestehende RPG-Anwendungen häufig nicht detailliert genug dokumentiert wurden und daher den verschiedenen Funktionen bei der Neuentwicklung nicht ausreichend Beachtung geschenkt wird. Dies kann bei der Implementierung dieser Funktionen erhebliche negative Auswirkungen auf den Prozessfluss haben.
- **Migration auf eine neue Umgebung.** Bei automatisierten Prozeduren wird die Funktionalität der vorhandenen Software vollständig oder schrittweise in eine moderne Entwicklungsumgebung übertragen. Risiken und Kosten lassen sich dadurch erheblich reduzieren. Im Verlauf des Transformationsprozesses können Qualitätsverbesserungen durch Umstrukturierungen oder Modularisierung der Struktur erzielt werden.

### Gründe für die Migration auf IBM Rational Business Developer

Als moderne Zielumgebung für geschäftskritische Anwendungen kommt Rational Business Developer und der integrierten Enterprise Generation Language (EGL) eine besondere Bedeutung zu. Durch die kosteneffiziente EGL-Codegenerierung wird die Erstellung von sich wiederholendem und gängigem Anwendungsinfrastrukturcode automatisiert. So lassen sich Fehler durch manuelle Codeeingaben weitgehend vermeiden und der gesamte Transformationsprozess wird verkürzt. EGL kombiniert die Stärken der Java-Technologie mit der leistungsfähigen Entwicklung von nativem Code für IBM i und System z Umgebungen. Die daraus entstehenden Anwendungen laufen daher praktisch auf jeder Plattform (Linux, Windows oder UNIX, IBM i oder IBM z/OS-Systeme) optimal.

---

## Highlights

---

Durch die Einbindung in Eclipse erweist sich EGL als moderne, serviceorientierte Sprache für die effiziente Entwicklung unterschiedlichster Anwendungstypen: Webprogramme, Datenbankanwendungen, Web-Services sowie Batch- und andere leistungsfähige Server für die schnelle Transaktionsverarbeitung.

Dank der engen Einbindung in die IBM Rational Tools lassen sich die EGL-Entwicklungsprozesse in andere professionelle Entwicklungsprozesse oder Lebenszyklen integrieren – dies reicht vom Anforderungsmanagement über die Modellierung bis hin zu Codierung, Test und Implementierung.

***EGL vereinfacht die Transformation von RPG-Anwendungen, indem Entwickler ihr Know-how in wesentlich kürzerer Zeit zu EGL als zu Java übertragen können.***

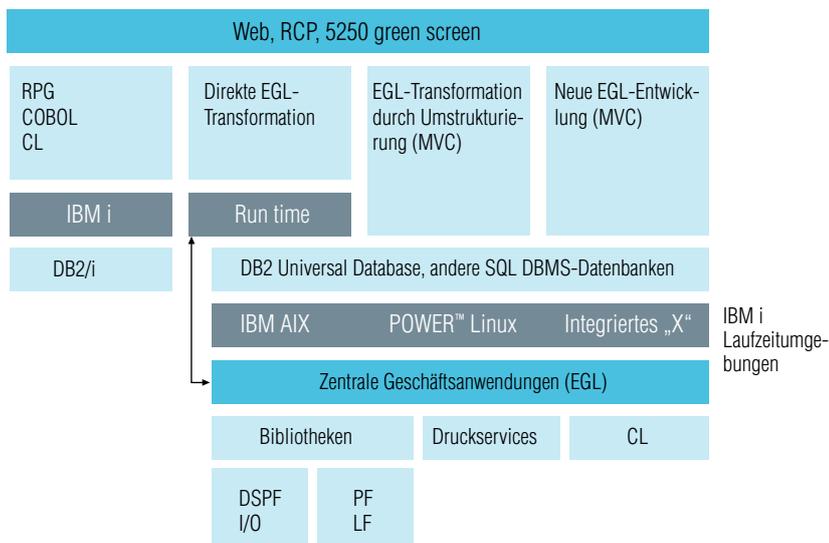
Die einfache Implementierung von EGL trägt dazu bei, die Transformation der RPG-Anwendungen zu erleichtern. Da EGL einfach zu erlernen ist, können RPG-Entwickler ihr Know-how im Vergleich zur Java-Technologie in einem Bruchteil der Zeit auf EGL übertragen.

**Highlights**

*Unternehmen können ihre aktuellen Verfahren und Tools zur Softwareentwicklung praktisch für jede Plattform modernisieren und verbessern, indem sie RPG-Anwendungen auf EGL migrieren.*

**Szenarios zur Anwendungstransformation**

Durch die Migration von RPG-Anwendungen auf EGL können Unternehmen ihre aktuellen Verfahren und Tools zur Softwareentwicklung praktisch für jede Plattform modernisieren und verbessern. Nachfolgend sind einige der gängigsten Szenarios aufgeführt.



Transformationsszenarios für IBM i Anwendungen.

Die Anwendung verbleibt auf dem Betriebssystem IBM i  
 Wenn die migrierte Anwendung nur auf dem Betriebssystem IBM i benötigt wird, können die vorhandenen RPG-Programme in einem einfachen, schrittweisen Prozess auf EGL migriert werden. Spezielle IBM i Funktionen wie Steuersprache, Befehle, APIs und Systembefehle werden in Anwendungen häufig eingesetzt. Diese Funktionen sind auch nach der Transformation verfügbar und machen somit einen Komplettaustausch überflüssig.

---

## Highlights

---

Die Transformation erfolgt in der Regel in zwei Phasen. In Phase 1 entkoppelt ein Server-Builder-Tool die vorhandene RPG-Anwendung von der 5250-Benutzerschnittstelle. Die RPG-Anwendung nutzt dann die API-Funktionen für den direkten Zugriff auf eine webbasierte oder Windows-Benutzerschnittstelle. Die digitale Datenspeicherquelle erstellt automatisch die GUI mithilfe einer GUI-Formatvorlage mit mehr als 60 neuen Funktionen. Mit sehr geringem Aufwand wird die Anwendung in eine Serveranwendung mit GUI transformiert, sodass interaktive Prozessorleistung entfallen kann.

In Phase 2 werden die RPG-Programme nacheinander zu EGL migriert. Die resultierenden EGL-Programme können dieselbe API für die Benutzerschnittstelle wie die RPG-Programme verwenden. Dank der einfachen EGL-Integration in RPG und die GUI lassen sich über EGL problemlos neue Funktionen hinzufügen und neue EGL-Programme direkt in einer SOA entwickeln. Programme, die aktualisiert werden müssen, werden in EGL neu generiert oder die bestehende Architektur wird per Refactoring in eine SOA transformiert.

***Über einen zweistufigen Ansatz werden die plattformspezifischen Funktionen und der Datenbankzugriff über RPG-Programme in den EGL-Bibliotheken gekapselt.***

Das Endergebnis sieht wie folgt aus: Die plattformspezifischen Funktionen und der Datenbankzugriff über RPG-Programme werden in den EGL-Bibliotheken gekapselt. Durch die Trennung der Geschäftslogik von der plattformspezifischen Schicht wird die Portierbarkeit von Anwendungen deutlich verbessert.

Die Anwendung wird auf einer Linux-, Windows- oder UNIX-Plattform benötigt. Da EGL plattformunabhängig ist, können migrierte RPG-Programme unter Umständen auch auf anderen Plattformen verwendet werden. Aufgrund der umfassenden Funktionalität des Betriebssystems IBM i sollten bestimmte Features etwas näher betrachtet werden. In Linux-, Windows- und UNIX-Umgebungen gibt es keine Standarddatenbank wie in der IBM i Umgebung. Daher wird eine Datenbanksoftware wie IBM DB2 benötigt.

---

## Highlights

---

***Wird eine Anwendung auf der IBM System z Plattform benötigt, können IBM i RPG-Anwendungen über zwei verschiedene, unterschiedlich komplexe Ansätze mithilfe von EGL auf die IBM System z Plattform migriert werden.***

RPG-Anwendungen, die Funktionen des Betriebssystems IBM i nutzen, welche in Linux-, Windows- oder UNIX-Umgebungen nicht zur Verfügung stehen, werden aus der Anwendung extrahiert und in einer speziellen Servicebibliothek bereitgestellt. Dies erfolgt manuell oder halbautomatisch über eine regelbasierte Umstrukturierung. Wenn auf diesen Plattformen kein RPG-Compiler verfügbar ist, muss die Anwendung vollständig auf EGL migriert werden, bevor sie ausgeführt werden kann. Mit den entsprechenden Tools lassen sich Daten aus der DB2-Datenbank in der IBM i Umgebung automatisch in die DB2 Universal Database-Umgebung übertragen.

Die Anwendung wird auf der IBM System z Plattform benötigt  
Mit EGL können vorhandene IBM i RPG-Anwendungen auf zwei Arten auf die IBM System z Umgebung migriert werden, wobei beide Verfahren unterschiedlich komplex sind.

Beim ersten Verfahren wird die DB2-Datenbank in der IBM i Umgebung auf die DB2-Datenbank in der IBM System z Umgebung migriert. Alle RPG-Stapelverarbeitungsprogramme werden ebenfalls auf EGL migriert. Die Befehlssprache CL von der IBM i Plattform wird entweder manuell oder halbautomatisch über regelbasierte Systeme auf die CL in der IBM System z Umgebung migriert. Interaktive RPG-Programme werden dann per EGL auf die Linux for System z Plattform migriert. Die DB2-Software wird ebenfalls auf der IBM z/OS-Plattform genutzt. Eine Webschnittstelle wird als Benutzerschnittstelle generiert, sodass die interaktiven EGL-Programme in den Java- und Linux for System z Umgebungen ausgeführt werden können.

Der zweite Verfahrensansatz setzt voraus, dass interaktive und Stapelverarbeitungsprogramme per EGL bereits auf die IBM System z Umgebung migriert wurden. Damit die interaktiven Programme mit der IBM CICS-Software kommunizieren können, müssen die RPG-/EGL-Programme restrukturiert werden. So kann ein Wechsel von der prozeduralen zur transaktionsorientierten Architektur erfolgen. Dies kann manuell oder halbautomatisch mithilfe regelbasierter Systeme vorgenommen werden und resultiert in einer nahezu reinen IBM System z Anwendung.

---

## Highlights

---

**Der Erfolg von IBM Rational Business Developer baut auf sechs Schritten auf, wenn es darum geht, eine effiziente Transformation von RPG-Anwendungen und langfristige Vorteile für das Unternehmen sicherzustellen.**

### Fazit

Für Unternehmen, die sich für IBM Lösungen entscheiden, um die Risiken bei der Modernisierung ihrer RPG-Anwendungen auf der IBM i Plattform zu minimieren, ergeben sich daraus – auch langfristig gesehen – zahlreiche Vorteile:

- *Automatisierte Codegenerierung zur Vermeidung von Fehlern durch manuelle Codeeingaben*
- *Weniger Zeit- und Kostenaufwand für die Anwendungswartung*
- *Risikominimierung durch Wiederverwendung und Erweiterung bewährter Funktionen und bestehender IT-Assets*
- *Vereinfachung der SOA-Unterstützung*
- *Mehr Flexibilität und Reaktionsfähigkeit durch konsolidierte Entwicklerteams, die je nach Geschäftsanforderung flexibel und projektorientiert eingesetzt werden können*

Die Modernisierung einer RPG-Anwendung ist ein strategisch wichtiger Schritt auf dem Weg in eine erfolgreiche Zukunft des Unternehmens. Folglich bedarf ein solcher Ansatz einer sorgfältigen Vorbereitung. Die Unternehmen können davon ausgehen, dass langfristig gesehen der Programmieraufwand eines großen Entwicklerteams zu zahlreichen unerwarteten Störungen führen wird. Um eine nahtlose Transformation sicherzustellen, baut der Erfolg von IBM Rational Business Developer auf sechs Schritten auf:

- *Aufbau von EGL-Know-how und Sammlung von praktischer Erfahrung in der EGL-Entwicklung*
- *Konvertierung eines bestehenden RPG-Programms in EGL zur gründlichen Überprüfung*
- *Durchführung einer umfassenden, detaillierten Analyse der bestehenden Anwendung(en)*
- *Durchführung einer Konzeptnachweisstudie für einen kleinen Teil der Anwendung*
- *Transformation der Anwendung mithilfe erfahrener Migrations-spezialisten von IBM und IBM Business Partnern*
- *Nutzung der Vorteile der neuen EGL-Funktionalität*



## Weitere Informationen

Wenn Sie mehr über IBM Rational Business Developer erfahren möchten, wenden Sie sich an Ihren IBM Vertriebsbeauftragten oder IBM Business Partner oder besuchen Sie uns unter:

[ibm.com/software/awdtools/developer/business](http://ibm.com/software/awdtools/developer/business)

IBM Deutschland GmbH  
Pascalstrasse 100  
70569 Stuttgart  
**ibm.com/de**

IBM Österreich  
Obere Donaustrasse 95  
1020 Wien  
**ibm.com/at**

IBM Schweiz  
Vulkanstrasse 106  
8010 Zürich  
**ibm.com/ch**

Die IBM Homepage finden Sie unter:

**ibm.com**

IBM, das IBM Logo und ibm.com sind eingetragene Marken der IBM Corporation. Power und Rational sind Marken der IBM Corporation in den USA und/oder anderen Ländern. Sind diese und weitere Markennamen von IBM bei ihrem ersten Vorkommen in diesen Informationen mit einem Markensymbol (® oder ™) gekennzeichnet, bedeutet dies, dass IBM zum Zeitpunkt der Veröffentlichung dieser Informationen Inhaber der eingetragenen Marken oder der Common-Law-Marken (common law trademarks) in den USA war. Diese Marken können auch eingetragene Marken oder Common-Law-Marken in anderen Ländern sein. Eine aktuelle Liste der IBM Marken finden Sie auf der Webseite „Copyright and trademark information.“ unter

[ibm.com/legal/copytrade.shtml](http://ibm.com/legal/copytrade.shtml)

Java und alle auf Java basierenden Marken sind Marken von Sun Microsystems, Inc. in den USA und/oder anderen Ländern.

Linux ist eine eingetragene Marke von Linus Torvalds in den USA und/oder anderen Ländern.

Microsoft und Windows sind Marken der Microsoft Corporation in den USA und/oder anderen Ländern.

UNIX ist eine eingetragene Marke von The Open Group in den USA und anderen Ländern.

Weitere Unternehmens-, Produkt- oder Servicennamen können Marken anderer Hersteller sein.

Der Inhalt dieser Dokumentation dient nur zu Informationszwecken. Obwohl die in dieser Dokumentation enthaltenen Informationen auf ihre Vollständigkeit und Genauigkeit hin überprüft wurden, wird sie auf der Grundlage des gegenwärtigen Zustands (auf „as-is“-Basis) ohne jegliche Gewährleistung zur Verfügung gestellt. Darüber hinaus basieren diese Informationen auf der aktuellen Produktplanung und -strategie von IBM, die sich jederzeit ohne Vorankündigung ändern kann. IBM übernimmt keine Haftung für irgendwelche Schäden, die aus der Nutzung dieser oder einer anderen Dokumentation entstehen oder damit in Zusammenhang stehen. Aus dem Inhalt dieser Dokumentation können kein Gewährleistungsanspruch oder andere Anforderungen an IBM (oder seine Lieferanten oder Lizenzgeber) abgeleitet werden, noch kann der Inhalt eine Änderung der Bedingungen der geltenden Lizenzvereinbarung, der die Nutzung der IBM Software unterliegt, bewirken.

<sup>1</sup> Gartner, „Gartner Says IT Leaders Must Place IT Modernization at the Core of Their 2008 Objectives,“ Pressemitteilung, 26. Februar 2008, [www.gartner.com/it/page.jsp?id=611507](http://www.gartner.com/it/page.jsp?id=611507)

<sup>2</sup> „Topic Overview: Legacy Applications,“ Forrester Research, Inc., August 2007.

© Copyright IBM Corporation 2009  
Alle Rechte vorbehalten.