

# DV-Dialog

IBM BUSINESS SYSTEMS • SYSTEM i • AS/400

Im Gespräch mit *Heidi Schmidt* und *Wilhelm Stolz* von der PKS Software GmbH, Ravensburg

## IT-Systeme mit RPG OA und EGL modernisieren



FOTO: DIETER KÖLBL

Viele Unternehmen investieren gezielt in die Entwicklung individuell angepasster Anwendungen statt in Standardsoftware, um sich von Mitbewerbern abzuheben und ihre internen Geschäftsprozesse optimal zu unterstützen. Auf dem System i sind diese Anwendungen in der Regel seit den 80er Jahren in der Programmiersprache RPG entwickelt worden. Während auch die ersten RPG-Anwendungen immer noch so ausgeführt werden wie früher, kamen mit neuen RPG-Versionen im Laufe der Zeit immer umfassendere und komplexere

Funktionen hinzu – getreu der Grundidee des Betriebssystems IBM i, Upgrades vornehmen zu können, ohne Anwendungsprogramme ändern zu müssen. Das bringt ein Dilemma mit sich: Die Funktionen, die für Stabilität in der Kundenumgebung sorgen, bringen gleichzeitig auch mehr Komplexität ins Unternehmen – und zwar durch Technologien, die im Verlauf der vergangenen 15 bis 20 Jahre jeweils den aktuellen Stand darstellten, heute aber eigentlich nicht mehr gebraucht würden. Dank der Kompatibilität der RPG-Versionen war der Kunde zudem in der Lage, im Lauf der Zeit stufen-

weise neue RPG-Funktionen zu implementieren. Daraus resultierte eine Mischung aktueller und veralteter Programmteile. Die technische Alterung, die den schleichenden Verlust der geschäftlichen Vorteile einer Software nach sich zieht, wird häufig jedoch erst dann festgestellt, wenn im Verlauf des Lebenszyklus unverhältnismäßig hohe Wartungskosten auflaufen oder die Anwendungsflexibilität nachlässt. Wie sich hier durch eine gezielte Anwendungsmodernisierung gegensteuern lässt, erläutern die beiden Experten Heidi Schmidt und Wilhelm Stolz von PKS Software.



# „Dauerbrenner Modernisierung!“

TEXT: BERTHOLD WESSELER | FOTOS: DIETER KÖLBL

Im Gespräch mit den beiden Modernisierungsexperten

*Heidi Schmidt* und *Wilhelm Stolz*

von der PKS Software GmbH, Ravensburg

**F**rau Schmidt, viele bestens bewährte Anwendungen sind mittlerweile in die Jahre gekommen. Woran erkennt ein IT-Leiter, dass er hier einen Modernisierungsbedarf hat?

**Heidi Schmidt:** Gute Frage. Hier möchte ich zunächst antworten, dass für wirklich jedes Unternehmen, das sich langfristig am Markt positionieren will, das Thema „Modernisierung“ von zentraler Bedeutung ist. Denn nur wenn ein Unternehmen mit der Zeit geht, rechtzeitig auf Trends reagiert und sich täglich mit den Herausforderungen der Märkte auseinandersetzt, wird es bestehen. Für mich ist Modernisierung daher kein reines IT-Thema.

Doch nun zur IT: Leider ist es häufig so, dass die IT- oder Entwicklungsleiter viel zu lange warten, bis sie das Thema Modernisierung der Software-Entwicklung angehen. Wenn ein Unternehmen heute noch Software im Haus entwickelt, steht dahinter ja ganz klar die Strategie, bewusst nicht auf Standardsoftware in bestimmten Bereichen zu setzen, um sich z.B. schneller und besser an Marktgegebenheiten anpassen zu können.

**Wilhelm Stolz:** Solch eine Unternehmensstrategie muss die IT-Abteilung professionell umsetzen; da gehört die „Modernisierung“ nicht nur alle Schaltjahre auf die Agenda. Sie muss vielmehr fundamentaler Bestandteil der IT-Strategie sein und sich in dieser nachhaltig niederschlagen. Wer dies nicht gewährleistet, wird sich eher früher als später kaum noch gegen die Ablösung der Individual-Entwicklung positionieren können.

Ziel muss also nicht ein Ampelsystem sein, aus dem man ableiten kann, wann mal wieder eine Modernisierung nötig wäre. Ziel muss es stattdessen sein, eine strategische Ausrichtung samt kurz-, mittel- und langfristiger Meilensteine auszuarbeiten und umzusetzen. Das ist der erste Punkt.

Der Zweite ist dann sicher, dass man sich natürlich genau überlegt, mit welchem IT-Partner und welchen Produkten und Lösungen man das Thema angeht, aber das ist, wie gesagt, erst der zweite Schritt. „Ein bisschen Modernisierung“ alle paar Jahre ist sicher vergeudetes Geld und verschwendete Liebesmüh.

**Neudeutsch ist die Rede von „Legacy Applications“, wobei Legacy wahlweise als Vermächtnis, Hinterlassenschaft, Erbschaft oder auch Alllast übersetzt wird. Wie kann man erkennen, ob es sich bei einer Applikation um ein wertvolles Erbe oder um historischen Ballast handelt?**

**Schmidt:** Liefert die Individual-Software Prozessunterstützung, die maßgeblich zum Unternehmenserfolg beiträgt, so ist dies ein idealer Bereich, in dem die firmeneigene Entwicklung Sinn macht. Dann gilt es zu prüfen, ob die gewachsene Anwendung auf einem wertvollen fachlichen Fundament steht.

**Stolz:** Ein Beispiel: Ist das Datenbankmodell schon nicht mehr dem heutigen Bedarf entsprechend, macht eine Modernisierung der darauf aufbauenden Programme keinen Sinn. Wenn die Anwendung dagegen über ein geeignetes Datenbankmodell verfügt, nur das Frontend aber nicht mehr zeitgemäß ist, dann ist eine Modernisierung meistens empfehlenswert und eine sinnvolle Investition.

**Schmidt:** Bei Prozessen, die wie etwa im Personal- oder Controlling-Umfeld branchenübergreifend standardisiert sind, macht es in der Regel keinen Sinn, an einer Eigenentwicklung festzuhalten. Es sei denn, es handelt sich hier tatsächlich um Kernkompetenzen des Unternehmens. Dies könnte beim Thema Personal zum Beispiel bei Personaldienstleistern oder ähnlichen Unternehmen schon der Fall sein.

*Bei alten Zöpfen gibt es im Prinzip zwei Wege für einen sauberen Schnitt bei der Modernisierung: Die Anschaffung einer Standardsoftware oder eine komplette Neuentwicklung. Wann ist letzteres zu empfehlen?*

**Schmidt:** Eine Neu-Entwicklung ist aufwendig und risikobehaftet. Schließlich sind die Bestandsanwendungen häufig über Jahrzehnte entwickelt und mit Funktionalität angereichert worden – das darf nicht unterschätzt werden. Doch natürlich ist es auch verlockend, auf der „grünen Wiese“ neu anzufangen und alte Zöpfe damit loszuwerden. Wählt man diesen Weg muss man in jedem Fall eine klare initiale Investitionsbereitschaft seitens der Geschäftsführung haben, um die Doppelbelastung aus Betrieb der Bestandsanwendung und Erstellung der neuen Anwendung zu tragen.

**Stolz:** Viele Offshore-Projekte der letzten Jahre haben zudem auch kritisch über diesen Ansatz nachdenken lassen, denn er birgt eben nicht nur die Chancen der „grünen Wiese“ sondern auch die Gefahr der „never ending story“, vieler Missverständnisse zwischen Entwicklern und Fachbereich und verzögerten Meilensteinen. Gerade dieses Spannungsfeld aus Chancen und Risiken einer Neu-Entwicklung ist aber unser Ansatzpunkt mit dem von uns entwickelten Konzept einer sanften, nachhaltigen und sicheren Modernisierung von Bestandsanwendungen. Dadurch sind Kunden in der Lage, vorhandene Assets mitzunehmen und trotzdem von den Vorteilen einer Neuentwicklung zu profitieren.

*Es gibt heute einen gut gefüllten Werkzeugkasten für die Programmierer, mit EGL, Java, PHP, Eclipse & Co. Welche Tools empfehlen Sie für mittelständische Unternehmen mit AS/400-Vergangenheit?*

**Schmidt:** Jeder mittelständische Betrieb ist individuell, insbesondere die IT- und Software-Landschaft. Daher wäre es recht unseriös, hier alle Firmen mit einem Lösungsansatz zu bedienen. Wir betrachten uns jeden



*»Screen Scraping ist nur dann eine Lösung, wenn die Anwendung stabilisiert ist, sprich nicht mehr weiterentwickelt wird.«*

**Heidi Schmidt**

Kunden und empfehlen dann firmenspezifisch auf Basis verschiedener Aspekte:

- wie entwickelt der Kunde heute?
- was wurde bereits ins User Interface investiert?
- welche fachlichen Anforderungen hat die IT in den nächsten 12 Monaten umzusetzen?
- wie sieht die Teamstruktur aus?

**Stolz:** Das ist nur ein kleiner Ausschnitt aus dem Fragenkatalog, den wir zusammen mit dem Kunden beantworten und auf dessen Basis wir ein passgenaues Konzept zusammenstellen. Schließlich ist das Thema viel zu kritisch, um hier mit Standardmodellen zu agieren. Kern unseres Ansatzes ist es aber immer, dass wir genau beleuchten, was von den vorhandenen Assets mitgenommen werden kann, da Investitionsschutz bei unseren Kunden meist ganz oben steht.

*Mit RPG Open Access modernisiert IBM jetzt ja auch ihre bewährte Programmiersprache für die Midrange-Welt. Wird damit auch der Klassiker RPG tauglich für Neuentwicklungen in Richtung Web 2.0?*

**Stolz:** RPG Open Access bietet eine definierte Schnittstelle und somit einfache Möglichkeiten, an klassische RPG-Anwendungen Web 2.0 Frontends „anzubauen“. Der Ansatz ist aber nicht wirklich neu; wir bei PKS nutzen genau solch eine Schnittstelle schon seit Jahren in unseren Produkten, die am Markt als Server Builder und Xi-Client bekannt sind. Kunden wie z.B. Dachser, Philips oder Voestalpine wissen deren Vorteile zu nutzen. Aber der Open Access Ansatz ist nicht klassisch für Neuentwicklungen im Bereich RPG-Anwendungen positioniert, sondern adressiert doch eher den Bedarf der Kunden, an vorhandene Anwendungen zeitgemäße Frontends anzuschließen – nicht mehr, aber auch nicht weniger!

*Worauf ist zu achten, wenn vorhandene Programme mit RPG OA modernisiert werden?*

**Schmidt:** IBM bietet ja zunächst mit dem Produkt RPG Open Access (5377-OA) lediglich den Unterbau bzw. die Basis. Darauf aufbauend benötigt man dann den sogenannten Handler sowie ein Frontend. Dies kann sich der Kunde entweder selbst schreiben, was in den seltensten Fällen wirklich Sinn machen dürfte, oder bei einem Anbieter kaufen. PKS beispielsweise bietet eine OEM-Lizenzierung für den Rational Developer for Power an, sozusagen ein Komplettpaket für die Open Access Umsetzung. Das Paket heißt RDP for RPG OA, ist bereits verfügbar und trifft auf breites Kundeninteresse.

*Wann sollte ein IT-Chef die Entwicklung eigener „Handler“ in Betracht ziehen, wann empfiehlt sich der Kauf von Produkten?*

**Schmidt:** Den Handler selbst zu schreiben sollte man gut überlegen. Schließlich muss er den kompletten Leistungsumfang des 5250-Treibers abdecken. Wie schon erwähnt, können wir bei PKS hierbei auf unser Produkt Server Builder aufsetzen, in dem mehrere Mannjahre Entwicklungsarbeit und einige hundert Kundeninstallationen stecken.

**Stolz:** Ich kann mir nicht vorstellen, dass es, insbesondere für Endkunden Sinn macht, etwas selbst zu entwickeln, was es bereits gibt. Das kann höchstens für Softwarehäuser Sinn machen, die ihre Produkte ja auch permanent modernisieren.

*Was sind die Stärken ihres Handlers?*

**Stolz:** Unser Handler ist bereits einige hundert Mal installiert, performant, hat einen maximalen Abdeckungsgrad und lässt sich völlig nahtlos in den Entwicklungsprozess beim Kunden integrieren. Durch die hohe Verbreitung können wir die Lösung preislich sehr attraktiv für Open Access anbieten. Last but not least möchte ich im deutschen Markt hervorheben, dass wir unseren Support ausschließlich aus Ravensburg heraus bedienen und somit keinerlei Sprachbarrieren zu überwinden haben.



»Starke Nachfrage sehen wir beim Thema Unicode-Unterstützung.«

Wilhelm Stolz

**Hat PKS auch Tools für Screen Scraping und Sprachanpassungen parat? Wann setzt man besser darauf – und wann sollte man in eine Programmänderung investieren?**

**Schmidt:** Reines Screen Scraping unterstützen wir nicht mit eigenen Produkten. Hier empfehlen wir HATS von IBM. Screen Scraping ist nur dann eine Lösung, wenn die Anwendung „stabilisiert“ ist, sprich nicht mehr weiterentwickelt wird. Denn sonst steht der Aufwand bei jeder Anpassung der Anwendung auf Dauer in keinem gesunden Verhältnis zum Gesamtaufwand, da diese Änderungen ja am Frontend nachgepflegt werden müssen. Somit ist das eine Variante, die doch eher seltener gewählt wird.

**Stolz:** Sehr starke Nachfrage sehen wir dagegen beim Thema Sprachanpassung bzw. Unicode-Unterstützung. Durch die zunehmende Verlagerung und Erweiterung der Geschäftstätigkeit nach Osteuropa und Asien steigt der Bedarf, auch die System-i-Anwendungen in diesen Ländern zu betreiben.

**Schmidt:** Der klassische Green Screen unterstützt dabei leider keine Darstellung von Unicode-Zeichen auf dem Frontend. Der Gerätetreiber nimmt hier vielmehr knallhart eine Konvertierung von EBCDIC nach ASCII vor und kann so „Datenschrott“ produzieren – ein riskantes Unterfangen, das alleine durch manuelles Agieren mit verschiedenen CCSIDs nicht vernünftig abzusichern ist.

Wir bieten hier eine einfache und effiziente Technologie, die es ermöglicht mit minimalem Aufwand auf Unicode umzusteigen, von der Datenbank bis zum Frontend. Dabei muss bei unserem Ansatz keinesfalls die gesamte Datenbank auf Unicode umgestellt oder gar DoubleByte installiert sein.

**Stolz:** Dann sprechen Sie noch das Thema „Programmänderung“ an. Dazu möchte ich

sagen, dass bei einer nachhaltigen Modernisierung immer im Einzelfall auch eine Anpassung von Programmen notwendig und sinnvoll sein kann, um langfristig fit für die Zukunft zu werden und auch zu bleiben.

**Was kostet die Modernisierung der RPG-Anwendungen mit OA – in Form von Lizenzen für Software-Produkte und in Form von Arbeit an den RPG-Programmen?**

**Stolz:** Eine Modernisierung mit RPG OA orientiert sich an der konkreten Umgebung des Kunden, da wir ja auf den vorhandenen IBM-Lizenzen aufsetzen. Pauschalaussagen sind hier also nicht möglich.

**Schmidt:** Wir können aber sicherstellen, dass wir in aller Regel eine betriebswirtschaftlich sinnvolle und akzeptable Lizenzierung darstellen können. Arbeit an den RPG-Programmen fällt bei unserer Lösung im Allgemeinen keine besondere an.

**Welche technischen Voraussetzungen auf Server-Seite sind für den Einsatz der RPG OA-Handler von PKS zwingend?**

**Schmidt:** Die Voraussetzungen für die Nutzung von RPG OA liegen in der IBM-OA-Schnittstelle begründet. IBM bietet die OA-Schnittstelle 5377 erst ab V6R1 sowie mit RPG LE an. Kunden, die dies nicht leisten können bzw. hier nicht upgraden können oder wollen, können aber unsere native Server-Builder-Variante nutzen, die wir ebenso als IBM-Produkt anbieten.

**Mit welchen Endgeräten bzw. Webservices arbeitet dieser Handler zusammen?**

**Welche Voraussetzungen sind hier zu beachten?**

**Schmidt:** Der Handler arbeitet mit Windows-, Web1.0- und Web2.0-Frontends zusammen. Für die Web-Welt benötigt der Kunde natürlich einen Webbrowser in sei-

ner Infrastruktur und ein entsprechend vernünftiges Netzwerk für seine Endbenutzer.

**Gibt es „Best Practices“, die Sie IT-Chefs bei der Einführung von RPG OA empfehlen?**

**Schmidt:** Wir empfehlen in jedem Fall eine Testinstallation samt Hands-On-Workshop, damit ein Kunde entscheiden kann, ob die Lösung den konkreten Bedarf abdeckt und wie die Entwickler mit den Produkten zu-recht kommen. Ebenso wird nur durch die Testinstallation klar, welche Ergebnisse erreicht werden können und wo auch die konkreten Grenzen dieses Ansatzes liegen.

**Mit EGL hat PKS auch eine Alternative zu RPG OA und Screen Scraping parat. Was sind die Vorteile bzw. typische Einsatzszenarien bei AS/400-Anwendern?**

**Schmidt:** EGL ist für all jene Kunden interessant, die langfristig nicht ausschließlich mehr auf RPG als Programmiersprache setzen möchten. EGL hat sich in den letzten Jahren gerade bei AS/400-Anwendern sehr häufig als bessere Alternative zu Java durchgesetzt, da es eine höhere Produktivität in der Software-Entwicklung bietet.

**Stolz:** Das liegt daran, dass Entwickler die Anforderungen des Business schneller umsetzen können als mit nativem Java. Und trotzdem kann man mit EGL die schon vorhandene RPG-Anwendung weiter als Basis nutzen.

**Warum sollte ein IT-Leiter auf EGL setzen, wenn er sich von RPG bei der Neuentwicklung verabschiedet? Mit Java und PHP hat er doch wesentlich populärere Alternativen...**

**Schmidt:** Klar, Java ist eine Alternative und sicher auch keine schlechte. Wer es sich als Kunde leisten kann, ein Java-Team neu aufzubauen oder in der Lage ist, die vorhandene RPG-Mannschaft auf Java umzupolen, wird dieses Alternative sicher in Betracht ziehen.

**Stolz:** Kunden, die sich für EGL entscheiden oder entschieden haben, tun dies aber ganz bewusst, weil sie sich eben nicht zwei parallele Teams leisten können oder wollen. Und von RPG auf Java umzusteigen schafft auch nicht jeder Entwickler.

**Schmidt:** PHP bietet mir persönlich zu wenig für die tatsächliche Entwicklung von Geschäftsanwendungen, da es für mich eher eine Skriptsprache ist. EGL dagegen ist ein pragmatischer Ansatz für Unternehmern, die effizient und mit einem prozeduralen Programmier-Background schnell in die Web2.0-Entwicklung einsteigen will und gleichzeitig vorhandene System-i-Assets wiederverwenden möchte. Klar mögen Java und PHP populärer sein – aber ob ich mit der Popularität auch meine Businessprobleme gelöst bekomme, muss ich selbst bewerten. Das System i ist auch keine populäre Betriebssystemplattform; dennoch geht die oftmals Todgesagte AS/400 nun schon stabil und zuverlässig durch das dritte Jahrzehnt ihres Produktlebenszyklus. ■