

APPLICATION MODERNIZATION

An RPG to J2EE Roadmap

By Phil Coulthard and George Farr, IBM Toronto Lab

Often we are asked from RPG and COBOL programmers, what they can do to modernize their tools, user interface, application logic and even their skills. They often ask if the end of a modernization road has any place for them and for RPG or COBOL. Many shops today have multiple camps, or teams, focused on different platforms and technologies. For instance, in addition to the iSeries development team, there is often another team that owns the company's public Web presence. The Web presence typically starts as simple static pages on Microsoft, Oracle or Sun technologies, created by people hired for that purpose, or as a result of outsourcing the project.

Inevitably the Web site evolves to require access to the iSeries logic and data, and then the turf wars begin. The Web team states they could easily re-write that old RPG stuff on Windows or Solaris, and take over the back office. On the other hand, the iSeries team knows this is a gross simplification and a case of extreme optimization born of young minds. Further, they have been hearing that the iSeries itself has far more capability than they are leveraging it for, including world-leading support for Java and Web applications. They wonder, would the opposite be true? Could in fact, the iSeries itself take over the Web work currently done on these other platforms, with these other technologies?

Well, we don't wish to take sides, except to say of course it can! The iSeries is in fact an awesome choice for "applications", be they Web or green. The truly exciting fact is that an existing RPG or COBOL shop can easily learn to leverage and master this power that is at their fingertips.

By doing so, they will spend more time in their CIO's office, but this time making their own proposals for the future IT strategy. Proposals that involve less cost, lower risk, and higher returns in reliability and performance, and even supporting the hot new "on demand" capability the press are buzzing about.

There are other, perhaps even more compelling, reasons for RPG and COBOL programmers to understand the Web capabilities of their favorite machine. This is because today, the preferred user interface for most appli-

cations, including in-house, is a Web user interface. Some applications still lend themselves better to a green-screen, and some are better suited to a richer client-server application style. However, the vast majority are best suited to a Web user interface that is more compelling than a 5250, and yet offers the same very simple distribution advantages. For client-server graphical user interface applications, the same type of discrete business logic design is needed for the iSeries business logic as a new Web application needs, and often this business logic can be re-used for either user interface. Thus, even for these applications, there is similar work and skills needed to get from where we are today to where we need to be, to build these types of applications.

For teams still using green-screen tools like SEU to build green-screen applications, there are many good reasons to consider something different.

IBM's strategy for applications and application development is consistent with many others in the industry ... use open standards and a large, rich, active developer community to build applications that are portable, resilient, and scalable. Use leading edge tools that are extensible and open, to build these modern applications.

Both this runtime, and these tools, are available to you as iSeries programmers.

The application framework for all of this is the Java 2 Enterprise Edition, or J2EE. It is from Sun Microsystems, but is the result of an open community

process with major contributions from all the industry players, including massive contributions and investments from IBM. The core of J2EE is of course Java, but it does not exclude the use of non-Java technologies as well, such as RPG or COBOL. Indeed, many successful J2EE applications on iSeries use RPG and COBOL for their business logic. J2EE can be crudely broken down into two distinct functional areas: the Web part, consisting of JavaServer Pages (JSPs), Java Servlets, and simple Java Beans; and secondly the logic part, consisting of Enterprise Java Beans (EJBs) and Message Driven Beans (MDBs). An application which uses only one of these two parts is still considered a J2EE application, and in fact we find most iSeries J2EE customers use JSPs and Servlets, but not EJBs. This is perfectly fine, and one of the things we hope to convey to you in the coming months is when the use of each technology is most appropriate.



George Farr

Phil Coulthard

It is our hope that almost every iSeries development shop will start building Web applications or at least well architected applications that can support multiple user interfaces, where today they build green-screen applications with hard-wired user interface logic. It is our job at the IBM Toronto Lab to ensure that every iSeries development shop has the tools needed to make it as easy to do this, as it is to use SEU and SDA to build traditional applications. Our strategy is not to do this via some massive shock therapy, but rather by incrementally expanding the comfort zone of RPG and COBOL developers. By supplying a new generation of tools that exceed what PDM, SEU and SDA do today, and bring SDA-like capability to Web development, and offer skills

re-use and even code re-use to some degree. Tools that support not only RPG and COBOL, but also Java, Web, XML, database, Web Services and, for those who need it, Enterprise Java Beans. The good news is that these tools exist today, and continue to improve rapidly. The outrageous news is that if you have RPG or COBOL, you probably already have unlimited licenses to these tools. They are yet another of the iSeries' awesome powers left untapped.

As of V5R1, the only way to purchase RPG or COBOL or SEU is to purchase WebSphere Development Studio for iSeries. This package includes all four Toronto compilers: RPG, COBOL, C and C++, and all the traditional green-screen tools, including PDM,

SEU, SDA, RLU and DFU. While the constituent pieces can be optionally installed, you can only purchase the one package, containing everything. Further, WDS entitles customers to unlimited licenses of WebSphere Development Studio Client for iSeries, which is the workstation-based next-generation of tools. There is one CD set for the client tools shipped, but it can be installed on any iSeries developer's computer for the purpose of doing iSeries application development. The current release of the client tools is V5, numbered to be consistent with WebSphere Application Server. The client tools are typically released more often than the server-side components. To get the latest V5 CDs, at no additional cost, simply order 5722-WDS, with feature code 5903.

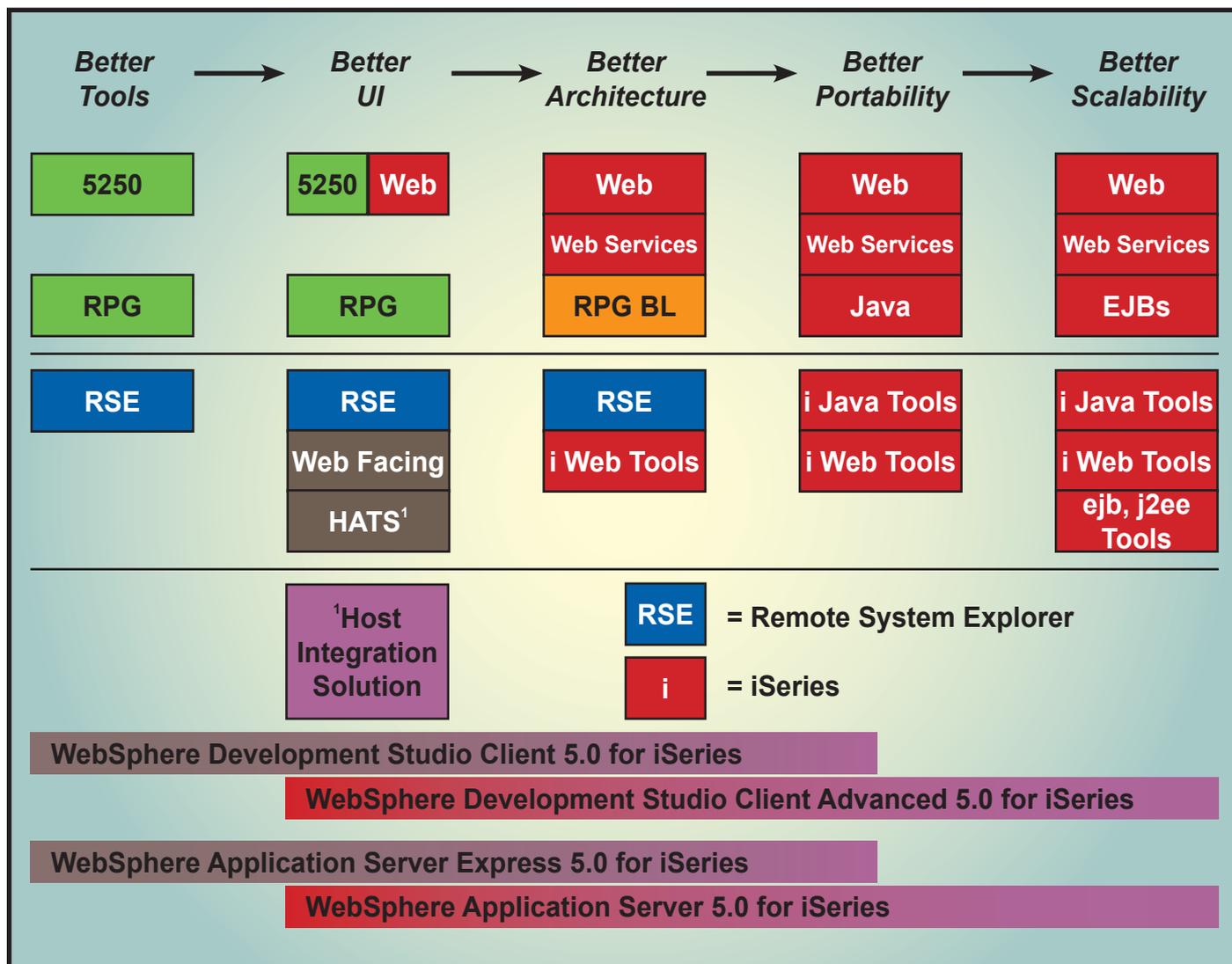


Figure 1. RPG to J2EE Roadmap

For those teams doing Enterprise Java Bean work, or requiring other advanced AD tools, there is a new WebSphere Development Studio *Advanced* for iSeries, which offers unlimited licenses of WebSphere Development Studio Client *Advanced* for iSeries. These client tools add support for EJBs and MDBs, and many other advanced features such as test case creation and management, and some advanced support in WebFacing and the Web tools.

This new packaging (relatively) solves one of the oldest barriers to modernization, the dreaded budget. If you have SEU, you also have the follow-on to SEU. So we now have the tools we need, but we need the skills to use them. Indeed, we also need the compelling *reason* to use them. We believe the reasons will become evident as we start exploring these tools and showing you their highlights, and the amazing things they will enable you to do. These are the secret weapons of the IT heroes. We'll want to make you a hero! What else do you need, beyond the tools? Well, once into building Web user interfaces, you will need WebSphere Application Server. You can't afford that? Well, that message was heard too, and the answer is WebSphere Application Server Express, a new smaller and significantly cheaper version of the fuller-powered WebSphere Application Server and WebSphere Application Server Network Deployment offerings. The Express version does not support Enterprise Java Beans, but it does support JavaServer Pages and Servlets. Combined with non-EJB Java, RPG or COBOL for the business logic, you have a powerful platform at an outrageously low price.

So we have the tools, and we have the runtime technology. What else do we need? We need a plan, Stan. We need a *roadmap* that will take us iteratively through the technologies, teaching us how to leverage and exploit the tools and technologies at each step. We offer here such a roadmap, which will be used as the framework for our subsequent articles as we walk you through the roadmap introducing the technologies and tools at each step.

In **Figure 1**, we see a roadmap for moving from RPG (or COBOL) and SEU, all the way to full J2EE, with Enterprise Java Beans and their tools in WebSphere Development Studio Client Advanced. Future articles will go into more detail on each step in the roadmap, but here we offer a quick high-level ("executive") introduction:

- The first step is simply to use **better tools** to do RPG and COBOL development. These tools are the Remote System Explorer, which is part of the WebSphere Development Studio Client, and are a follow-on, and super-set, of PDM, SEU, SDA, RLU and the system debugger. They will increase programmer productivity, and help with attracting and retaining the best developers.
- The second step is to add a **better user interface** onto an existing application, using non-invasive technologies like Host Access Transformation Server (HATS) or WebFacing.

RESOURCES

For more information on modernizing applications, check out the following resources:

- For information on this road map, including education available, see the iSeries Enablement Assistance Web site (www-1.ibm.com/servers/enable/nav2/i/index.html)
- For information on the iSeries WebSphere* Development Studio Client (DSc), see the DSc Web site (www.ibm.com/software/awdtools/wds400)
- The article "Something For Every Developer" is available online (www.eservercomputing.com/iseries/articles/index.asp?id=607). It contains an overview of new functions available with DSc 5.0.
- The article "GUI Catches Fire" is available online (www.eservercomputing.com/iseries/articles/index.asp?id=660). It contains information on IBM's WebFacing and Host Access Transformation Server products.
- Several other articles on related topics can also be found on the **IBM iSeries edition eServer magazine's** Web site (www.eservercomputing.co) by using the Search function.

— P.C.

HATS transforms the 5250 datastream at runtime, while WebFacing converts your display file source into JSP source, and intercepts the application I/O at runtime, before the creation of the 5250 datastream. HATS is good for immediate Web-enablement of entire sessions, while WebFacing is good for evolving a particular application. Both allow iteration of resulting Web pages. WebFacing Tools are part of WebSphere Development Studio Client, while HATS tools come with the Host Integration Solution product.

- The third step is the biggest one yet, involving creating a new Web application or application module that has a **better architecture** (model-view-controller) and discrete business logic. Typically this non-interactive logic is created from scratch, but it can also be culled from existing source via clever use of copy and paste. The iSeries Web Tools in WebSphere Development Studio Client are used for these applications, generating the non-RPG controller logic and even the first pass at the JavaServer Pages user interface. While the tools do not remove the need to learn dynamic HTML and JSP technology, they can significantly defer it, just as SDA does for DDS. Further, the tools allow those with Web skills to work along side those with the RPG skills. 

- The fourth step is for those who need **better portability**, to more of the eServer family say. This means using non-EJB Java to write the business logic. Another reason for this move would be to gain the benefits of Object Oriented development that Java offers. To fully exploit OO, someone on the team will learn to use analysis and design tools (OOAD) and technologies, such as offered by Rational. The WebSphere Development Studio Client toolset is built on top of Eclipse (www.eclipse.org) so it has extraordinary Java tooling built-in, although OOAD tools are separately purchasable, plus comes with iSeries-specific Java tools for exporting and importing, and remotely compiling, running and debugging, for example.
- The fifth step is for those who need **better scalability** in their applica-

tions, such as for public web sites, or simply wish to fully exploit all the power of a full J2EE application. These applications use Enterprise Java Beans, Message Driven Beans and many other J2EE technologies that will enable highly scalable, fully portable, applications. The EJB and other advanced J2EE tools come with WebSphere Development Studio Client Advanced.

Of course, there are other important technologies such as commerce and portals that we leave to you to explore.

On the other hand, we also do not want to imply that everyone needs to follow the roadmap directly, or go all the way to EJBs. Where you start and where you end, and even how you move from step to step, is situation dependent.

We offer this as a guideline only, and a framework for our future discussions.

We hope you will come with us on the journey. To prepare, maybe you should look for those WebSphere Development Studio Client CDs, or place an order for your no-additional-cost copy of them.



George Farr and Phil Coulthard are co-authors of the Midrange Computing books *Java for RPG Programmers* and *Java for S/390 and AS/400 COBOL Programmers*. George can be reached at farr@ca.ibm.com, and Phil can be reached at coulthar@ca.ibm.com.

IBM, WebSphere, and iSeries are trademarks of International Business Machines Corporation in the United States, and/or other countries. Java is a trademark or registered trademark of Sun Microsystems, Inc. in the United States, and/or other countries. Windows is a trademarks of Microsoft Corporation in the United States, and/or other countries. Other company, product, and service names may be trademarks or service marks of others.

This article was reprinted with permission, from "eServer Magazine, iSeries edition", published by International Business Machines Corporation, Rochester, MN

Solutions Beyond Limits...
eServer has IT all!
April 20 - 21, 2004
Sheraton Parkway Hotel
 The Toronto Users Group for Midrange Systems (TUG) once again presents the annual Technical Education Conference and Showcase (TEC). We will feature the latest hot topics in the eServer realm and the topmost expert speakers. For more information contact the TUG Office at 905-607-2546 or e-mail: admin@tug.ca.

TUG
TEC2004