IBM Revitalizes System i App Dev Tools in V6R1

by George N. Farr

IBM's Rational software has revitalized its i5/OS application development tools and compilers with changes in the areas of product packaging and rebranding, new client products, enhancements in RPG IV V6R1, and the introduction of the newest i5/OS business language — EGL. The latest tools, which were announced by IBM in late January, and as of this writing, are scheduled to roll out on March 21, 2008.

Although the changes are substantial, they are not complicated. The purpose of this article is to offer a high-level summary of what was announced in V6R1 for WDS and the Eclipse-based client products for Version 7.1. We will publish future articles relating to other enhancements announced in V6R1.

George Farr

At a Glance

IBM has introduced three new features of WDS; ILE Compilers, Legacy Compilers, and ADTS tooling, with flexible pricing and packaging enabling System i shops to consume only the pieces / parts they require. The new announcement also includes major new functions in RPG IV/ COBOL and workstation tools.

Three new products are also introduced: Rational Developer for System i (RDi), Rational Developer for System i for SOA Construction (RDi SOA) and HATS for 5250.

IBM's newest business language — EGL — which will be part of Rational Developer for System i for SOA construction (RDi SOA), will help System i shops extend their existing RPG and COBOL applications to the web.

Our ultimate goal with the changes we are bringing to you in this announcement is to ensure that we enable our customers and business partners to protect client investment in applications running on i5/OS, provide continuing support in the foreseeable future, and enable customers to modernize their applications with modern Web and Web 2.0 User Interfaces.

Packaging and Rebranding

Customers reported for years that the WDS one–size–fits–all model didn't work. Customers want to consume only the product components that they need. The new packaging clearly identifies our strategic products versus heritage products to encourage customers to make the right choices. A side benefit, this new packaging will help IBM clearly see which features are being used and therefore will impact our decisions for future development investment.
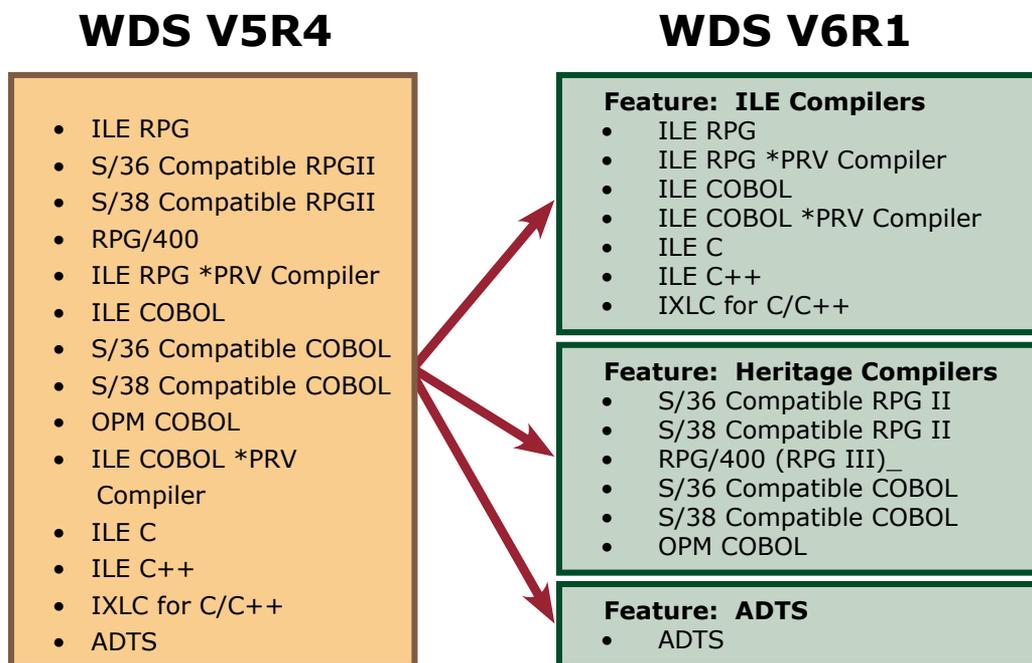
WDS V5R4

- ILE RPG
- S/36 Compatible RPGII
- S/38 Compatible RPGII
- RPG/400
- ILE RPG *PRV Compiler
- ILE COBOL
- S/36 Compatible COBOL
- S/38 Compatible COBOL
- OPM COBOL
- ILE COBOL *PRV Compiler
- ILE C
- ILE C++
- IXLC for C/C++
- ADTS

WDS V6R1

Feature: ILE Compilers
- ILE RPG
- ILE RPG *PRV Compiler
- ILE COBOL
- ILE COBOL *PRV Compiler
- ILE C
- ILE C++
- IXLC for C/C++

Feature: Heritage Compilers
- S/36 Compatible RPG II
- S/38 Compatible RPG II
- RPG/400 (RPG III)_
- S/36 Compatible COBOL
- S/38 Compatible COBOL
- OPM COBOL

Feature: ADTS
- ADTS

Figure 1.

The new WDS (Product ID 5761–WDS) will now contain three separate components:

ILE Compilers — including RPG, COBOL, C, and C++. IBM has made significant enhancements to them and will continue to do so.

Heritage Compilers — including S/36 RPG II, S/38 RPG II, RPG/400 (RPG III), S/36 COBOL, S/38 COBOL, and OPM COBOL. These products are stabilized. This implies that IBM will continue to offer support and service for them but will no longer enhance them.

ADTS (SEU, PDM, SDA, RLU, DFU) — This product will continue to ship but will also be stabilized with no future enhancements.

Figure 1 illustrates the current WDS packaging in V5R4, which contains all the compilers and tools, and the corresponding new packaging under V6R1 that unbundles WDS into three different features.

Although IBM has priced each feature separately, the company has also implemented a user-based pricing model with a reasonable number of user licenses included for each machine's software tier level.

The WDSc Standard Edition will no longer be included with the WDS product.

Websphere Development Studio client (WDSc) as well as Websphere Development Studio client Advanced edition (WDSc AE) have now been stabilized; therefore, IBM will place all future System i application development enhancements and investments in our new product offerings. Marketing will end during the current quarter, and service will end in April 2010.

One point to note here, the application diagram as well as screen designer, originally only included in WDSC AE, are now included in our base offering in RDi.

Entitlement

Whenever we present these major announcements, the first question that comes up all the time is *"I am now on V5R3 or V5R4 and would like to upgrade to V6R1, how does this work and am I entitled to these products?"*

So let me address this question before moving on with further details of the announcement: If you are an existing customer and you have SoftWare Maintenance Agreement (SWMA) with IBM, the answer is yes, you will be entitled to a specified number of licenses for ILE compilers, Heritage compilers and ADTS in WDS. For example, if you currently have RPG or SEU it means that you are licensed for WDS. In addition, if you are on SWMA, you will then be entitled to 'x' number of licenses of ILE Compilers, Heritage compilers and ADTS tools.

As mentioned earlier, WDSC 7.0 has been stabilized, however, customers currently having it and wishing to continue using it can order it through IBM or their business partner. It is included in WDS as a supply feature. A supply feature in a product is used in the fulfillment center at IBM to deliver the product (or supply it) to you.

Furthermore, if you have WDSC AE, you are also entitled to both RDi SOA as well as Rational Application Developer (RAD).

What about RDi and RDi SOA, are you entitled to these products? These products are brand new and therefore you will need to purchase them first. You can also buy a maintenance agreement for them that will enable you to receive future releases of the products

New pricing model

At this point, you may wonder why is IBM moving toward a user based pricing model versus tier based? If you indeed wondered that, you are a smart person, reward yourself, go get a cookie and make sure you come back and continue reading ☺

IBM's old pricing model (Prior to V6R1) was tier based. That implied that you paid a price for WDS that is based on the tier you are on. A tier is the actual machine you are using, or the model number, this could be a P05, P10, P20 or all the way up to P60. The price of WDS ranged from few thousands dollars for the low tier, to above $100K for

the large tiers. Determining the value for the compilers, however, was in many cases difficult– as the size of the box has little relation to the usage of the compilers or developers using those compilers.

Our new pricing should provide a fair model for the value customers are receiving, and is similar to how compilers are packaged and priced across our hardware offerings.

Boost Your RPG or COBOL Skills with EGL

IBM Rational Business Developer (RBD) Version 7 Release 1 offers a number of enhancements which will let System i developers effectively leverage existing investments in RPG to produce modern, professional, web or SOA applications using their existing business logic quickly and with minimal time to market.

The RBD product provides a language called EGL, an easy–to–learn, high-level business language that you can use to modernize your applications quickly. RBD provides the development environment (including edit/compile/ debug capabilities), a web page creation editor, wizards, and dialogs that make easy to write applications in EGL and use RPG or COBOL business logic or data on i5/OS for web or SOA applications. The language and the tooling are a powerful combination that IBM believes will make your organization more productive and let you modernize your applications in a cost-effective manner

You can use EGL and RBD wizards and editors to easily create web applications that can use your business logic (written in RPG or COBOL) and business data on i5/OS. Your existing investments are protected.

You can also write new business logic in EGL or continue to develop business applications in RPG and/or COBOL, which can be modernized for the web or SOA as explained above. The advantage of using EGL is that you can generate Java or COBOL code from it and run applications on different platforms and middleware (applications servers, databases, and so on). In other words, you can develop platform-independent business applications using EGL.

As a high-level language, EGL hides the complexity of writing web applications using Java 2 Enterprise Edition (J2EE), which can be difficult to master, and dealing with the complexities of deployment on your middleware (application server settings, setting up data sources for use, and so on). The EGL language syntax is also closer to RPG and COBOL than Java, which is object-oriented and can be complex. If you are an RPG or COBOL developer, you can learn EGL much faster than other languages such as Java. EGL, like RPG and COBOL, is a procedural language andd uses record level I/O. The best part is that the tooling in RBD lets you not only edit but also debug your EGL code directly, so you never have to deal with the generated languages (Java or COBOL).

In addition, IBM is working on EGL Rich UI, an EGL extension that will be released as Tech Preview first and in a product in the future. With EGL Rich UI, you will be able to mashup and create composite web applications with rich user interfaces using the latest Web 2.0 technologies. (**See Figure 2.**)The intent is to leverage EGL's existing values of simplicity, abstractions, and portability to increase skills transfer, productivity, and time to market.

The RBD tooling will let you compose web pages in editors using rich UI widget libraries such as DOJO, YUI, and EXT and debug and trace applications from within the tool. You will be able to quickly write end–to–end Web 2.0 applications that use Ajax techniques for responsive web UIs. The applications you create can have mashups, feeds, and allow social or community-oriented input such as tags and recommendations. The ricu UI's can be easily integrated with back end RPG or COBOL processing.

Rational Developer for System i (RDi)

System i customers have long asked for a more lightweight tooling that focuses on the work at hand. Shops have reported that WDSc and WDSc AE are too big, contain features they are not using, and cause a slow
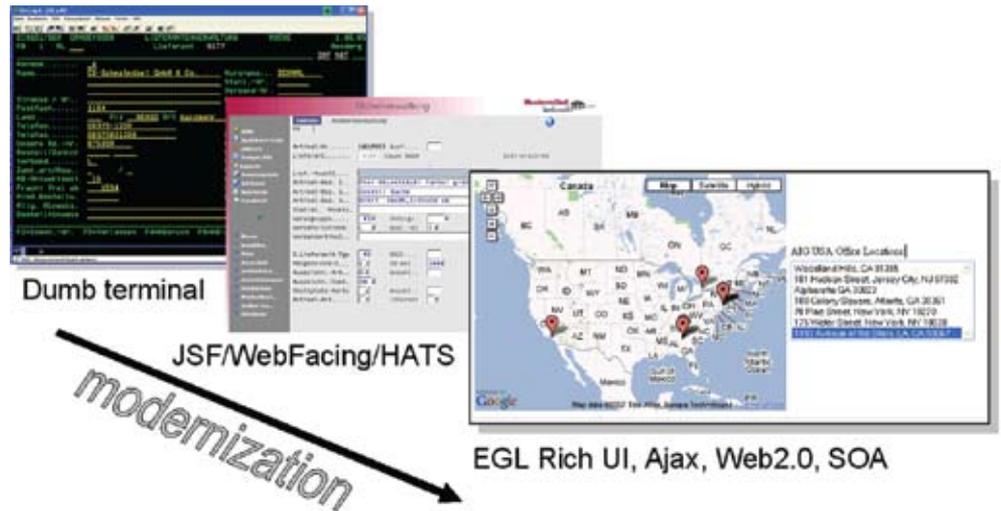

Figure 2.

start-up time. The new lightweight Rational Developer for System i (RDi) tool offers significant enhancements to let System i customers improve their application development. IBM's eclipse based answer to ADTS is RDi! (Product id: on the configurator is '5733RDi, and on passport advantage is 5724T82)

Like all other strategic IBM tools, RDi is based on the Eclipse platform and is exceptionally light with minimal resource requirements. It is certified to run on a desktop machine with as few as 512 MB of memory. IBM has focused on the needs of RPG, COBOL, and CL developers and is offering high-performance edit, compile, and debug capabilities along with access to i5/OS resources and facilities.

In the case that you never used the Remote System Explorer (RSE) which was included in WebSphere Development Studio Client, now RSE is included in RDi and offers you many powerful functions compared to the existing CODE/400 product as well as ADTS.

The new features of RDi include:

Full RPG and COBOL syntax checking and program verification that allows you to test compile your members locally without generating an object and reports the errors through the Error List view. You can then with a click of the mouse imbed the errors in the source member at the location they occur for easy fixing.

Content assist that allows you to display help anywhere in your source member. For example, if you are on a D specification and want to get a list of keywords that are allowed to be entered, you simply key Ctrl+Space and that will display the list of allowable keywords. In a column oriented language such as RPG, content assist comes very handy.

Outline view allows you to display the structural elements of the source member that is currently open in the editor. For example, for an ILE RPG source member, the outline contains all the structures, fields, subroutines, procedures indicators and so on. This comes in real handy as you modernize your large monolithic applications

Ability to display copy members by simply moving the mouse over the '/copy' statement and issuing a command.

You can also go to any File specification for an externally described file, right click the mouse, and select to see the details of the file. In a view, you will now see all the records and fields defined in the file.

Our latest release, RDi 7.1, also comes with many major new enhancements.

Application diagram was first introduced in version 7.0 and helped shops visualize their applications using capabilities such as call graphs showing subroutine, procedure, and program calls.

To see an application visually, you select resources by right clicking and selecting Visualize Application Diagram. You will then be presented with a graph as shown in **Figure 3**.

You can select any native members, that includes programs (*PGMs), service program(*SRVPGMs), IFS files, local files on your machine, and iSeries i5/OS project resources.

If you happen to have the source member open in the LPEX editor, you can also do the same command by right hand click the mouse and select > Visualize Application Diagram.

Application diagram helps you in modernization by expediting the turning of existing RPG applications into service programs and let users participate in SOA. The new application diagram in RDi brings additional capabilities, such as:

Functional zoom, which is the ability to start with top-level resources and drill down to see lower-level details as desired.

Program call support, which is represented as a connection from the calling subroutine or procedure node to the called program.

CL support by way of application diagram.

In addition, RDi includes the second phase of the screen designer to use for 5250 screen design directly in RDi. All the functionality of the screen designer and the application diagram that originally was available in WDSc AE is now available in RDi.

RDi 7.1 comes with more new enhancements. It offers V6R1 language support for RPG, COBOL, and CL; Remote Systems LPEX Editor with language parsers, syntax checkers, program verifiers, and RPG content assist and wizards; online language help; error list view enhancements; outline view enhancements; and new find-date in the editor.

RDi will not contain the web development tools or other features previously included in WDSc. This tool is built specifically for RPG, COBOL, and CL developers. In addition, RDi is user-priced so that you

Figure 3.

can buy as many or as few copies of the product as you need. RDi integrates with other Rational products, giving additional features and functions. The key to this is the fact that you add these features as needed. *System iNEWS* will include coverage of these specific enhancements in future issues.

Rational Developer for System i for SOA Construction (RDi SOA)

RDi SOA is designed for the developer who needs to build web applications, web services, or functionality beyond basic RPG, COBOL, or CL development. Note that RDi SOA is a software bundle that contains two sets of interrelated capabilities. RDi SOA is a combination of RDi and Rational Business Developer (RBD) with Enterprise Generation Language (EGL). It is intended for developers who want to leverage and extend existing RPG or COBOL applications to the web or to an SOA-based environment.

RDi SOA provides web development support with the Web Design tools and JSF support in the product. It includes the web service wizards and tooling required to enable ILE RPG or COBOL programs to implement or consume web services.

RDi SOA Product id on the configurator is '5733SOA', and on passport advantage it is '5724T83'.

WDHT Is Now HATS for 5250 Applications

For the last several years, IBM has been consolidating its two presentation-integration (refacing) technologies: WebFacing and HATS. A common runtime and tier-based pricing structure called WDHT and introduced in 2006 let customers take full advantage of both technologies. In 2007, the concept of "interoperability" was introduced so that the technologies could work together. In this release, IBM will be rebranding, renaming, and repackaging the common runtime as Rational Host Access Transformation Services (HATS) for 5250 Applications.

The WebFacing Tool, which has traditionally shipped as part of WDSc and WDSc AE, will be an optional installable component in the HATS Toolkit. The fully functional HATS Toolkit will still be available for free download from the web and will also ship with RDi SOA. After customers are ready to deploy their HATS applications to production to either reface or deliver a web service, they simply purchase the appropriate number of HATS licenses for 5250 applications.

IBM will also introduce several functional enhancements. With the HATS technology, you can create web applications that provide screen transformation access to your host applications from mobile devices. With HATS for 5250 Applications, you

can develop HATS portlets that comply with the standard Java Portlet Specification (JSR 168) API. You will be able to update web services even easier with the next release. Included with that release will be a technical preview of a Visual Macro Editor, a tool that lets the developer visually develop macros. Finally, the new WebFacing Application Bridge will let applications built with the WebFacing component transfer control and data to and from other web applications, such as EGL, giving you the ability to combine HATS, WebFacing, and EGL into a single application.

What's Happening to RPG IV and COBOL?

The biggest enhancement IBM added to RPG IV in March is a true multiple threading capability that is native to the language. Multi-threading gives you the ability to run in multiple threads and therefore allocates each thread its own static storage. You simply indicate THREAD(*CONCURRENT) on the control specification of a module to enable this function. Multiple threads can run in the module at the same time, and by default, static variables will be defined so that each thread will have its own copy of the static variable. This makes them thread–safe. You can define individual variables and indicate that they be shared by all threads using STATIC(*ALLTHREAD). If you are familiar with Java, this is similar to defining class variables. These variables are not thread–safe, by default.

Another major enhancement is the ability to define files locally in procedures. After the P specification you can now define one or more files. Input/Output to local files can only be done with data structures. In addition, there are no I and O specifications for local files. Note also that by default, the storage associated with local files is automatic; the file is closed when the subprocedure returns normally or abnormally. You can use the STATIC keyword to indicate that all invocations of the procedure will use the same open file. If the file is open when the procedure returns, it will remain open for the next call to the procedure.

Another enhancement is the introduction of the MAIN keyword on the control specification. It designates one subprocedure as being the main procedure that gains control when the program is called. In addition to being the program-entry procedure, the main subprocedure is like any other subprocedure. It does not use the RPG cycle. The prototype for the main subprocedure must have the EXTPGM keyword; the main subprocedure can only be called by a program call.

IBM has also added enhancements related to the use of files, which will come in the form of keywords on the F specification. The new enhancements relate to qualified formats, the LIKEFILE keyword, a new template keyword, passing files as parameters, EXTDESC to specify the file to be used at compile time, and an allow-data structure in the result field for EXTFMT.

Other added enhancements include significantly higher limits for the size of variables (from 64K to 16MB), a growth in the number of elements in an array from 32K to 16MB, an increase in the length of string literals from 32K to 16 MB, the relaxation of some UCS-2 rules (available starting in V5R3 through PTFs) the reduction of some module sizes have been reduced, and the ability to store parameter information in the program.

Stay Tuned

To prepare for the future, you continue to modernize your RPG IV code to make it more modular, use procedures, avoid mixing business logic with user interface, and take advantage of EGL. IBM is adding enhancements in RDi and RDi SOA that you have been asking for. Expect to see great enhancements in RPG IV in the future to further integrate it with EGL to provide you with better web experience. **TUG**

George Farr is the World Wide product line manager for IBM i tools and compilers. George is an award-winning speaker and author of Java for RPG Programmers, Java for S/390 and AS/400 COBOL Programmers, as well as ILE: A First Look and RPG IV by Example. He can be reached at farr@ca.ibm.com.