SIMPLIFY IBM i APPLICATION DEVELOPMENT WITH SMARTER PRODUCTIVITY TOOLS

BY MIKE STEGEMAN, SR. DATA ACCESS CONSULTANT

A SLOW DEVELOPMENT DEPARTMENT—ONE RELYING ON REDUNDANT PROCESSES OR LACKING VISIBILITY INTO ITS ENVIRONMENT—CAN BRING AN ENTIRE ENTERPRISE TO A GRINDING HALT. THEREFORE, IT IS IMPORTANT TO GIVE DEVELOPERS THE RIGHT TOOLS—SMART TOOLS—THAT IMPROVE PRODUCTIVITY BY HELPING THEM ACCOMPLISH TASKS IN AS FEW STEPS, AND WITH AS LITTLE ROOM FOR ERROR, AS POSSIBLE.

SEQUEL Software™

True Data Empowerment

SIMPLIFY IBM i APPLICATION DEVELOPMENT WITH SMARTER PRODUCTIVITY TOOLS

BY MIKE STEGEMAN, SR. DATA ACCESS CONSULTANT

Development environments on Power Systems™ running IBM i can be a challenge to learn, control, and document. Changes in source code and personnel are just two examples of the hurdles you must clear in order to gain and maintain control of programs and files.

Members of the development team need an easy way to monitor the applications they're working with, as well as track changes made by team-mates. Their managers must be able to track the team's progress accurately, and report on developments as required. Managers interact with auditors, who expect proper documentation to be available, and who may impose penalties if it is not. End users, of course, just want their programs to work.

A slow development department—one relying on redundant processes or lacking visibility into its environment—can bring an entire enterprise to a grinding halt. Therefore, it is important to give developers the right tools—smart tools—that improve productivity by helping them to accomplish tasks in as few steps, and with as little room for error, as possible.

This Is Your Life

Let's explore a few common development challenges and the questions they often raise.

Your development staff is working feverishly to keep pace with the program changes your users require. Longer days and shorter lunches are the new normal. Your stress level, and that of your team, is skyrocketing. How can you streamline workflows in your department?

You recently installed IBM Rational Developer for System i (RDi) to allow your developers to write and change source code from a PC GUI environment. The developers love it, but something is missing. Team members are constantly leaving the RDi environment to chase down a cross-reference or object reference, and then returning to RDi to make the necessary code changes. When the amount of code is large and the cross-reference is elusive, major slowdowns result. How can you improve the efficiency of your staff?

You and your staff have been asked to write a new system for tracking inventory as it moves through your warehouses, in addition to the multitude of program fixes and changes on your plate already. Is there a simple way for your designers and programmers to keep tabs on the programs and screens being created for the new package?

The CEO informs you that your company is acquiring another company. They have a payroll software package that is better than the one you currently use. She wants you to learn the new software and its data flow so you can support it and train the HR staff. How can you and your staff familiarize yourselves with the HR package in a timely manner?

Good news: You now have approval to hire the developer you've been looking for. After a moment of elation, you realize that while your new developer will arrive next week, he will need several months to get up to speed on your development environment, internal systems, and applications. How can you speed the transition process for new developers?

Internal corporate auditors have arranged a visit. They need you to show them the object relationships of the sales order system and how data flows from file to file, so they can report on the integrity of the information. How can you give auditors the information they need without diverting your team members from their current tasks?

Let's explore the effects these situations can have on your development environment, and the ways in which productivity solutions that include cross-referencing and documentation aids can dramatically improve your situation.

Development at the Speed of Light

Today's business world evolves at an unprecedented pace. Computer programs capture data at almost every level of business. Analysts evaluate data and use it to create business intelligence, which hopefully leads to new business opportunities. These opportunities in turn require more (or different versions of) data. This new cycle of data transference requires analytical users to receive data through different mediums and in various formats. Consequently, IT teams are increasingly relied upon to provide mission-critical information.

In large IT development departments where programs are constantly being changed, tweaked, and created, there is a strong need for an intuitive cross-reference tool. Even smaller development environments will benefit from increased productivity, especially when thousands of programs need attention.

For example, developers often spend inordinate amounts of time searching for programs that use a particular file or field. Even highly experienced

> # IT teams are increasingly relied upon to provide mission-critical information.

developers complain that they spend most of their time researching where to make changes and determining what impact their edits might have on other programs.

Imagine if, instead of hunting, your team members spent the bulk of their time making actual changes. The developer's time will have been used more effectively, and the end result will be available to users faster. A tool that can quickly show a developer all of the programs that use a file or field is an invaluable resource.

Supercharged RDi

The number of development departments "going rational" is growing. IBM RDi is a set of integrated development tools based on Eclipse. It allows developers to create, maintain, and enhance applications on Power Systems running IBM i. While RDi is a powerful tool, it has its limitations. As your developers multitask, they may need to leave RDi to determine object relationships. Likewise, an analyst will need a system overview in order to integrate new applications or programs into your existing environment. And managers and auditors want complete documentation of an application or environment.

> ## "It is a capital mistake to theorize before one has data."
> — Sir Arthur Conan Doyle

To have the best of both worlds, you need a tool designed for multiple users that integrates with RDi. Developers should be able to find valuable relationships, then edit the source, compile, and debug efficiently—with one program. The same tool should give analysts the up-to-date information about application structure, data flow, and object references they need to efficiently design changes. This same tool should document and create flowcharts for high-level analysis of system flow and object dependencies.

Shorter Learning Curves

Development productivity solutions can help provide a clearer path, whether you are creating a new program, learning to use a program, or helping someone else get acquainted with an application or environment.

When designing and writing a new application, you need to track and document such information as program flows, data flows, and object dependencies. It is imperative that the data you rely on in your work is current. Too often, developers spend valuable time making changes to applications, unaware that other members of their team have been making changes at the same time. It is critical for programmers to be able to keep track of their own progress, as well as the progress of others.

Similarly, when learning a new application you need visibility, whether it is a high-level processing overview or a detailed object reference and data flow process. The right tool will show you object references and object usages, and provide flowchart documentation. Productivity solutions that allow you an easy overview of an application, along with its underlying mechanisms, help you become proficient faster, meaning you will have more time to focus on your primary duties.

Moving from new applications to new employees, everyone knows that a great deal of the cost involved with bringing on a new member stems from training—it often takes months to bring a new hire up to speed. Therefore, sensitive budgets and the rapid pace of business require that team members be trained in as quickly as possible. Cross-reference and documentation tools will have them in the know and performing confidently in less time.

Happy Auditors, Happy Managers

Having someone pore over your work, looking for the smallest, out-of-place detail causes most of us anxiety. However, having another set of eyes review a process and point out potential loopholes is often worth the inconvenience, as it allows you to correct errors and avoid major problems down the road. It is always better to correct a process before it is rolled out to the user community than after it has created problems for your business.

Auditors and managers depend on good development productivity tools to provide them with this type of data flow documentation. The information they use must be current and accurate. When managers have good data, they are able to make wise, confident decisions about current and future projects. With good data, they can schedule employee time, as well as their own, more efficiently. In some cases, governmental regulations require a documented process of the data flow to be accessible or on-hand.

Choose Your Weapon

You should not experience undue stress while choosing a cross-reference and documentation tool, but neither should you take the process too lightly. Take

STAKEHOLDERS	WHAT'S AT STAKE
Developers:	*Require insight into their work and that of their teammates in order to avoid redundant work and programming errors.*
Managers:	*Must track team progress and provide documentation to auditors and executives, so as to avoid penalties, missed deadlines, and more.*
Auditors:	*Tasked with monitoring development to ensure adherence to corporate, governmental, or other regulatory policies.*
End Users:	*Rely on applications to carry out everyday job functions.*

time to scrutinize each of a solution's features and benefits. Compare costs, but also weigh them against losses you may already be incurring due to inefficient processes. A strong argument can often be made for investing in development productivity tools by putting a price tag on wasted or redundant work hours.

Of course, you could avoid costs by using a free solution. Freeware can be compared to public transportation in a metropolitan area—let's say Chicago. In relation to the cost of driving a car, including insurance and gasoline, the cost of riding the train is low. And, in most cases, it isn't difficult to catch a train going into or out of the city. But if you want to travel from a western suburb to a northern suburb, you will find there is no direct route. Your trip duration increases because you must ride into the heart of Chicago in order to catch a northbound train. This is an inefficient use of time. Assuming that time is an important factor, in this scenario it would be wiser to drive.

Like a cheap train ride, free productivity tools sound attractive until you consider all of the angles. If a tool does not meet the needs of your users, the tool truly has no value. If development efficiency does not improve, or if it declines, the tool is a speed bump at best and a roadblock at worst.

Shared Benefits

Just as you might use a development tool to gain a complete perspective on an application that needs fixing, use your knowledge and experience to study the needs and pain points of your organization. Look from all sides, from high points and low. Ask yourself who in your organization would benefit from a tool with the functionality outlined in this report. Consider exactly how the tool would help, and how those benefits might ripple into other areas and departments.

Chances are, you will discover that your developers could improve their efficiency. They likely devote too much time to performing manual tasks in order to get information that they could otherwise obtain almost instantly. And they would certainly benefit from having all relevant data available in one place.

You may also find that your auditors, who may have taken you to task for incomplete documentation in the past, would be pleased to have complete documentation for their reports available where and when they need it. Likewise, managers would get peace of mind from an improved ability to track what their teams are working on and to provide updates to the executive team.

When you add satisfied end users to the list of developers, auditors, managers, and executives who would benefit from productivity tools that emphasize cross-reference and documentation, it becomes evident that virtually everyone in your organization has a stake in improved efficiency. Therefore, with a keen understanding of where you are and the drive to reach your development goals, all that is left for you to do is choose the right tool.

Try ABSTRACT, the IBM i development cross-reference and documentation solution from SEQUEL Software.

Contact us for your free trial:
1-800-328-1000 | info@sequel-software.com

SEQUEL Software™
True Data Empowerment