

INTRODUCTION

What a ride! During the last couple of years, we have seen the rise and fall of many e-businesses, Web sites, and Internet gizmos. While some have been able to hang on for the long run, others were flashes in the pan. It has truly been a very interesting new millennium for the business world as well as the computer programmer, no matter what platform you work on.

The AS/400 has also seen a lot of changes. First and most noticeable is the name change from “Application System/400” to “the e-server iSeries 400.” Several enhancements came along with the name change, including the introduction of new Web tools and the porting of Tomcat, Linux, and the Apache Web server to the AS/400.

What do all of these changes and new technologies mean to the RPG programmer? To me, it shows that the strong will survive and evolve. RPG is the biggest survivor of them all. Even more, e-RPG has proven to be a very viable long-term solution for your e-business needs. Despite what some critics will try to make you believe, e-RPG is a proven solution that allows you to create fast,

upgradeable Web solutions for your AS/400 (I mean, iSeries 400). Those who have used e-RPG as a part of their e-commerce solutions will agree.

A VISIT TO THE PAST OF e-RPG

In early 2000, my book *e-RPG: Building AS/400 Web Applications with RPG* was published. This book steps through the basics of Hypertext Markup Language (HTML), JavaScript, configuring your AS/400 as a Web server, and the APIs used with RPG to write fully functional e-business applications. Applying these techniques, the reader can create fully functional Web applications using the tried-and-true RPG programming language.

Initially, I wasn't quite sure what response a book on e-RPG would get. After all, everywhere you turn, you're shown new programming tools, languages, and techniques that promise to do everything, and some even claim to do it better than native CGI programming languages such as RPG. Actually, though, it should have been no surprise that e-RPG was embraced with open arms as a solution to the "e-World." It was just what the doctor ordered.

One of the highlights during the publication of that book was an online e-RPG chat that I participated in. Many of the participants' questions had to do with specific solutions to specific problems, and others had to do with more general questions about the future of RPG. Since some pundits had said that new development was no longer taking place with RPG, I asked the participants if anyone was doing new RPG development. The response was that *all* were developing new applications in RPG and e-RPG and were proud of it, proving once again that what goes on in the real world is not necessarily what the media claims.

I received many emails thanking me for writing that book. I also received many success stories that included the use of e-RPG. Each time a reader emailed me, excited about getting a first e-RPG program up and running, it brought me back to the day when I did the same. It is a great feeling of accomplishment and of breaking down the false barriers put up around the "green" world. It also opens up the endless possibilities that wait in the future of e-RPG development!

I'd like to thank all of the readers who shared their success stories with me, no matter how big or small. It shows that e-RPG is to the Web what RPG is (*not* was) to the green screen and reporting world: the right tool for the job.

A VISIT TO THE FUTURE OF E-RPG

The general enthusiasm for the topic of e-RPG was immense. Many requests for a follow-up were made. This book answers those requests by taking you further into the depths of e-RPG. It introduces you to some new ideas, and also re-visits and improves upon some old techniques.

One concern I did have with the previous book was that the introduction to the Integrated Language Environment (ILE) might have been a bit short. This new book devotes an entire chapter to ILE. This chapter is based on the conference notes, demonstrations, presentations, and articles that I have written for training others with ILE.

I encourage you to study and use ILE, especially in your e-RPG applications, but also in your everyday applications. In a nutshell, ILE lets you write your own Built-In Functions (BIFs), as well as encapsulate APIs and everyday processes into small, manageable pieces. And you probably thought only Java could encapsulate! (Cue laughter.)

Also included in this book are advanced chapters on both HTML and JavaScript. These chapters include some tricks and techniques that are well worth sharing with everyone. And because HTML is the true backbone of any Web page or programming language, it is one subject you never can learn too much about.

Some readers of the previous book requested more information about setting up HTTP servers. In this book, I expand on this topic, as well show you new configuration directives that allow you to create logs, protect from a Denial of Service (DoS) attack, and set up custom error pages.

Another large change is that I no longer use dynamically built Open Query File (OPNQRYF) statements. Instead, I use dynamic SQL embedded in my e-RPG programs. This is, in my opinion, the single most important enhancement introduced

in this book (in addition to ILE, of course). The power of dynamic SQL is truly important. Once learned, it can be applied to many different situations and platforms, not only in e-RPG programs.

THE “HELLO WORLD” SYNDROME

What would a book be without a small piece of opinion from the author? Well, it would be a book free of opinion. (Again, cue laughter.) The topic of this opinion is what I call the “Hello World” syndrome.

You probably already know where I’m going with this. Every time we are shown a new technique or programming language, we are introduced to it with a program that outputs “Hello World” or something similar. I do it myself, in my previous book and in this one as well.

We tend to look for Hello World examples as a starting point because we are eager to produce output. The Hello World program usually cures our initial craving for conquering something new, but therein lies the possible problem.

By curing the craving for output, the Hello World program can give a false sense of security. I know I have been guilty of thinking I had actually done something when all I did was output some text to a screen. This sense of security becomes false when you realize that the main goal of learning a new technique or language is to do something functional for yourself or your company, not just to say “Hello” to the world.

I received a few emails from readers of my last book who had gotten my Hello World example working, but were having problems performing other, more complex tasks, such as reading data from a browser or formatting data from their databases into HTML tables for presentation. At the same time, I received emails from others who got their companies’ intranet and/or Internet applications up and running with fully functional sites that took orders, reported order statuses, and performed other, more complicated tasks.

My main point here is that it seems we still feel like we can’t afford the time to sit down and learn. I think it’s exactly the opposite. You can’t afford *not* to sit

down and learn. Being told that you are in charge of getting your company's intranet or Internet applications up and running, but that you don't have time to take a week or two to learn the necessary techniques, is ironic at best. It's like saying there's never time to do the job right, but there's always time to do it over.

Don't get me wrong: Hello World examples are a good starting point, and I think we should continue with them (although maybe change the output to something like "Look what I did, Mom!"). Don't stop there, thinking you've learned it all, however. The rush you receive when your Hello World program works is great, but use it as a catapult to more involved topics, not as an end in itself.

RIGHT TO THE POINT

The main point of my first e-RPG book, which remains the point of this book, is that you shouldn't be afraid to apply what you already know to new technologies such as Web programming. You don't have to totally ignore all the hype going on in the world—by all means, go out and experiment with what is out there. Don't forget, though, about the faithful machine, programming language, and (most importantly) programmers who have brought you and your company to where you are today.

Even if you decide that RPG is not the Web programming language for you, remember that you still have a very valuable team in your RPG programming staff. They not only know and understand the business rules that must be applied to the Web, but they also keep systems in the back-end running smoothly.

Remember that the Web front-ends must interact with the existing back-end systems. Before your front-end team starts building Web applications that will interface with your legacy applications, have it meet with the back-end team, so that they can help each other. Nothing is more frustrating than being nearly done with a project and realizing that the Web interface is missing important data needed by the back-end. (Don't ask me how I know that this *will* happen.)

Remember, not everything is Web-based, although with today's media hype, it might seem that way. And, just for the record, when I use the term "legacy," it is a proud compliment, not a putdown or an attempt to demean. Think about it: if an

application lasts long enough to be a legacy, that means it's worked. You probably can't say the same thing for a lot of software these days.

So, the next time you're reading your favorite periodical or news flash, before coming to the conclusion that the "buzzword of the day" is going to solve all of your problems, consult with your team to get their response. If approached in the right fashion, you might be surprised at the information you receive. You might even get some real-life experiences, not just "Hello World" hype.

ON WITH THE SHOW

It is now time to move onward and upward. In the following chapters, I share with you new techniques that I have learned and implemented in my personal e-RPG applications, as well as ideas sent to me by customers and readers of my first book.

Since the publication of that first e-RPG book, a lot has changed, and a lot has stayed the same. I encourage you to take your time and digest each new piece of information. Don't be in a rush. This will only hinder your learning and your learning experience.

Now, let's get on with the show!