Introduction

In the old days, our beloved server was, in many cases, the only one used by the company to perform its business. That's no longer true.

IBM i is no longer an island.

This book is about building bridges that link your IBM® i system to the modern business server world. It will show you easier and more flexible ways to get data into IBM i, along with rather surprising methods to export and present the vital business data it contains. It will help you automate file transfers, seamlessly connect PC applications with your RPG programs, and much more.

Your input operations will become more flexible and user-proof (even though, of course, a totally user-proof system is just a myth), with self-correcting import processes and direct file transfers that work a minimum of user intervention. Your DB2® data will look great on program-generated Microsoft® Excel® spreadsheets and browser-based, interactive charts, ranging from simple bar charts to complex charts with data tables.

All the solutions presented here are either based on existing open source tools or created from scratch by the author. The full source code is included (via download from the book's page at *http://www.mc-store.com*), along with sample programs that are easy to understand and adapt. This will help you integrate these input/output

methods into your own applications, providing flexible, integrated, and modern solutions. All of this is done with RPG Integrated Language Environment® (ILE) programs, avoiding the increased complexity of new servers and programming languages.

Upon completing this book, you will:

- Have new ways to link your IBM i system to the outside world
- Have learned how to automate boring tasks, like file transfers
- Be able to create, read, and write files on the IFS from an RPG program
- Know how to prevent and correct user errors on CSV import files
- Be able to create beautiful interactive charts directly from your RPG code, without having to learn a new programming language
- Be able to easily integrate Microsoft Windows®-based applications with your IBM i programs
- Have your RPG programs produce professional-looking spreadsheets instead of ugly printouts
- Have gained a set of open source, free tools that will help you solve everyday problems with ease and style
- Be inspired to expand what you've learned, creating your own solutions for better output

Book Structure and Organization

This book is divided into two parts: "Flexible Input" and "Dazzling Output." The first part explains how to automate data transfers and make text file uploading to your DB2 database more user-friendly and less IT-dependent, thus making your inputs more flexible. "Dazzling Output" might seem an overstatement, but the second part of the book contains ideas that will surprise you. Hopefully, it will have you thinking, "Wow, I never thought I could do that with only RPG!" It includes Excel file creation as a way to (easily) replace those ugly printer files that still lurk around your application (yes, we all have some), among other things.

This book also includes a bonus part, "Going Global." It includes two chapters about geo-referencing entities in your database and collecting geo-related information for your company and its clients' benefit.

Each chapter is divided into several parts, but should be read as a whole. It's also important to mention that some chapters are linked, because they use what

was discussed before as a basis for something new. Each chapter begins with an introduction, which includes its highlights and what you should know to fully enjoy it (knowledge of a certain technology/language, familiarity with material in other chapters, and so on).

Naming Conventions

The source code contained in this book follows a few naming conventions:

- File names:
    - All the physical file names start with *PF*.
    - All logical file names start with *LF*.
- Variables:
    - *W_* identifies a work variable.
    - *P_* identifies a parameter variable
      (usually a parameter of a procedure or function).
    - *K_* indicates that the variable is part of a key list.
    - *I_* indicates that the variable is an indicator (Boolean) variable,
      usually containing **On* or **Off*.
    - *C_* is used for constants.

Generic Compilation Instructions

The source code presented in this book is composed mostly of procedures, neatly contained in modules, which in turn are part of service programs. In order to get the sample programs to work, you might have to recompile some or all of the source code.

To compile RPGLE modules, use this command:

```
CRTRPGMOD MODULE(<MODULE_LIB>/<MODULE_NAME>) SRCFILE(<MODULE_LIB>/QRPGLESRC)
DBGVIEW(*SOURCE)
```

To compile SQLRPGLE modules, the following command should be used:

```
CRTSQLRPGI OBJ(<MODULE_LIB>/<MODULE_NAME>) SRCFILE(<MODULE_LIB>/QRPGLESRC)
SRCMBR(*OBJ) COMMIT(*NONE) OBJTYPE(*MODULE) DBGVIEW(*SOURCE)
USRPRF(*OWNER) DYNUSRPRF(*OWNER)
```

To compile service programs, this is the appropriate command:

```
CRTSRVPGM SRVPGM(<SRVPGM_LIB>/<SRVPGM_NAME>) EXPORT(*ALL)
TEXT(<SRVPGM_TEXT>) BNDDIR(<SRVPGM_LIB>/<SRVPGM_NAME>) ACTGRP(<SRVPGM_LIB>)
OPTION(*DUPPROC *DUPVAR) ALWLIBUPD(*YES) USRPRF(*OWNER)
```

To compile programs, use this command:

```
CRTPGM PGM(<PGM_LIB>/<PGM_NAME>) BNDDIR(<PGM_LIB>/<PGM_NAME>) ACTGRP(<PGM_LIB>)
OPTION(*DUPVAR *DUPPROC) ALWLIBUPD(*YES) USRPRF(*OWNER)
```

Open Source Tools Used in This Book

This book includes a lot of code that was created from scratch, but there are some cases in which it didn't make sense to reinvent the wheel, so to speak. In those instances, I resorted to a handful of open source software and expanded upon its functionality. Here's a list of what is used throughout the book:

- *Mime and Mail (MMail)*—This open source utility (library MMAIL) for IBM i allows you to create, send, and receive MIME (Multipurpose Internet Mail Extensions) files. It is authored by Giovanni B. Perotti, a brilliant Italian programmer. The software is frequently updated, so I recommend that you download the latest release and read the documentation from the tool's Web page:

  http://mmail.easy400.net

- *CGIDEV2*—Another of Giovanni's great tools, CGIDEV2 was actually created by Rochester IBMer Mel Rothman back in 1996. Giovanni took over a few years later and has maintained the tool ever since. This tool facilitates the communication between a Web page and your IBM i system. In several chapters, it's used to invoke REST (Representational State Transfer) Web services, so it's a good idea to download the latest version and read the documentation. You can find it here:

  http://cgidev2.easy400.net

- *POI-HSSF*—This pure Java® open source implementation of Excel file creation is used extensively in chapter 6. Don't be scared by the "pure Java open source implementation" bit. Scott Klement, another of the people I look up

to in the IBM i world, wrote a set of procedures that facilitate the use of this tool without knowing the least bit of Java. Be sure to read the documentation from Apache's POI Web site and any articles that you might find online about POI, because this is a very powerful and useful tool. You can download the necessary source code from Scott's POI page:

http://www.scottklement.com/poi

● *IFS tool*—Scott also wrote a nice little tool that makes reading and writing IFS text files from RPG much easier. I use it in chapter 5 to fix some common user errors found in CSV files. This is also a free tool, which you can download here:

http://www.scottklement.com/rpg/ifs.html