

# Why migrate to ILE RPG IV?

The Report Program Generator(RPG) language was designed by IBM in the early 1960s for the primary purpose of cranking out batch reports. Its built-in cycle and column oriented syntax seemed strange at first, but once the shock of this radically different coding style was overcome, many programmers learned to love the RPG language. Even though RPG had a loyal following, as the IBM midrange computer became more popular and business applications became more complex and interactive, the language was feeling quite restrictive.

Therefore, to satisfy today's AS/400 programmer, IBM released ILE RPG(also known as RPG IV) as a more functional and flexible business programming language. There has been a lot of hype over the last few years concerning this new release of RPG, and rightfully so. We must concur with the old-timers when they claim this is the most significant change they've seen to the RPG dialect yet. IBM has been listening!

While there are more enhancements to the RPG language than we have room to list, we will indicate some of the primary features you may be interested in:

- Ability to participate in the Integrate Language Environment(ILE), which allows you to break your large programs into small reusable modules without performance implications. We highly recommend using modular programming techniques to reduce your development and maintenance costs.
- Lower case support throughout all the specifications. Luckily, its not case-sensitive like the "C" language, so the variable names of "WORK1" and "work1" refer to the same variable. For readability, we recommend placing as much of your code as possible in lower case.
- Data structures, work fields, arrays, and constants are now defined in the new "D"efinition specifications. The "I"nput specifications were definitely becoming strained. We recommend defining all of your work fields in the "D"efinition specifications. You should not define any field lengths in your calculations.
- Many of field limitations have been either expanded or eliminated. Field names have been expanded from 6 to 14 characters; unfortunately, you now have to think of a whole new set mnemonics for naming your fields. The size of character fields has been expanded from 256 to 32767. The number of decimal places for a numeric field has been expanded from 9 to 30. The size of data structures has been expanded from 9999 to 32767. The size of constants has been expanded from 256 to 1024.
- In order to accommodate the new operation codes and longer field names in the calculation specifications, factor1, factor2, and the result field columns have been expanded to 14 characters and the operation code column has been expanded to 10 characters.
- Twenty characters of comments can be entered on the right side of every specification, as long as you create your source file with a length of 112.

- There is enhanced support for date and time fields. Many new operation codes have been added to convert, add, and subtract from date/time fields. You no longer need to use data structures or multiply by 10000 something to convert your dates. This would be a good time to consider using date and time type fields in your database, versus creating numeric fields to hold these values.
- The dynamic EVAL operation code has been introduced, which allows you to assign a value to a variable using a free form expression, which can span multiple lines.
- Since there is now more room in the operation code column, IBM decided to decrypt some of its existing operation codes. BITOF has been renamed to BITOFF, CHEKR to CHECKR, COMIT to COMMIT, DEFN to DEFINE, DELET to DELETE, EXCPT to EXCEPT, LOKUP to LOOKUP, OCUR to OCCUR, REDPE to READPE, RETRN to RETURN, SELEC to SELECT, SETOF to SETOFF, UNLCK to UNLOCK, UPDAT to UPDATE and WH to WHEN.
- Several new operation codes have been introduced with ILE RPG. ADDDUR is used to add intervals to a date or time. CALLB is used to call an ILE bound procedure. EVAL is used to assign a value to a variable through a free-format expression(discussed earlier). EXTRCT is used to extract a segment(i.e. the month) from a date or time. SUBDUR is used to subtract intervals from a date or time. TEST is used to determine if a date or time is valid.
- The operation codes of DOU, DOW, IF, WHEN, OR, and AND can now use free form expressions for compares in the extended factor2. For instance, instead of using \*IN90 IFEQ \*OFF, you can now use if \*in90 = \*off.
- There is now built-in pointer support for you API enthusiasts.
- Several new built-in functions(BIFs) have been introduced to the calculation specifications for determining field sizes, trimming off blanks, etc.
- The number of subroutines in a program has been expanded from 256 to unlimited.
- The number of elements in an array has been expanded from 9999 to 32767.
- The maximum number of files in a program has been expanded from 50 to unlimited.
- File name lengths have been expanded from 8 to 10 characters. It always seemed strange that IBM let you create a file with a 10 character name that couldn't be used in an RPG program without an OVRDBF.

As you can see, a multitude of enhancements has been made to the RPG language. We estimate a learning curve of 2 to 4 weeks for an existing RPG programmer to become proficient in ILE RPG.

By using Linoma's RPG Toolbox, you will watch how your RPG III and RPG/400 source code is translated into ILE RPG, which should dramatically accelerate your learning curve.