

REPORT MARCH 2023



RPG DEVELOPMENT

WHAT IS GOOD OR BAD IN RPG DEVELOPMENT?

Over the years we worked with RPG developers and companies owning very large RPG codebases. We have seen how RPG software is behind software playing a vital role in many organizations: sometimes is the language used to create the main products of these companies, other times is software used internally to ensure that the company runs properly, but in all those cases it is providing a lot of value to the organization who owns it.

Yet, what we noticed is that the community around RPG is not as vital as the value of RPG codebases would suggest. There are basically no open-source libraries for RPG, very few articles, or events around this language. RPG developers appeared to us to operate in silos, with little occasions to compare their experiences and learn from each other.

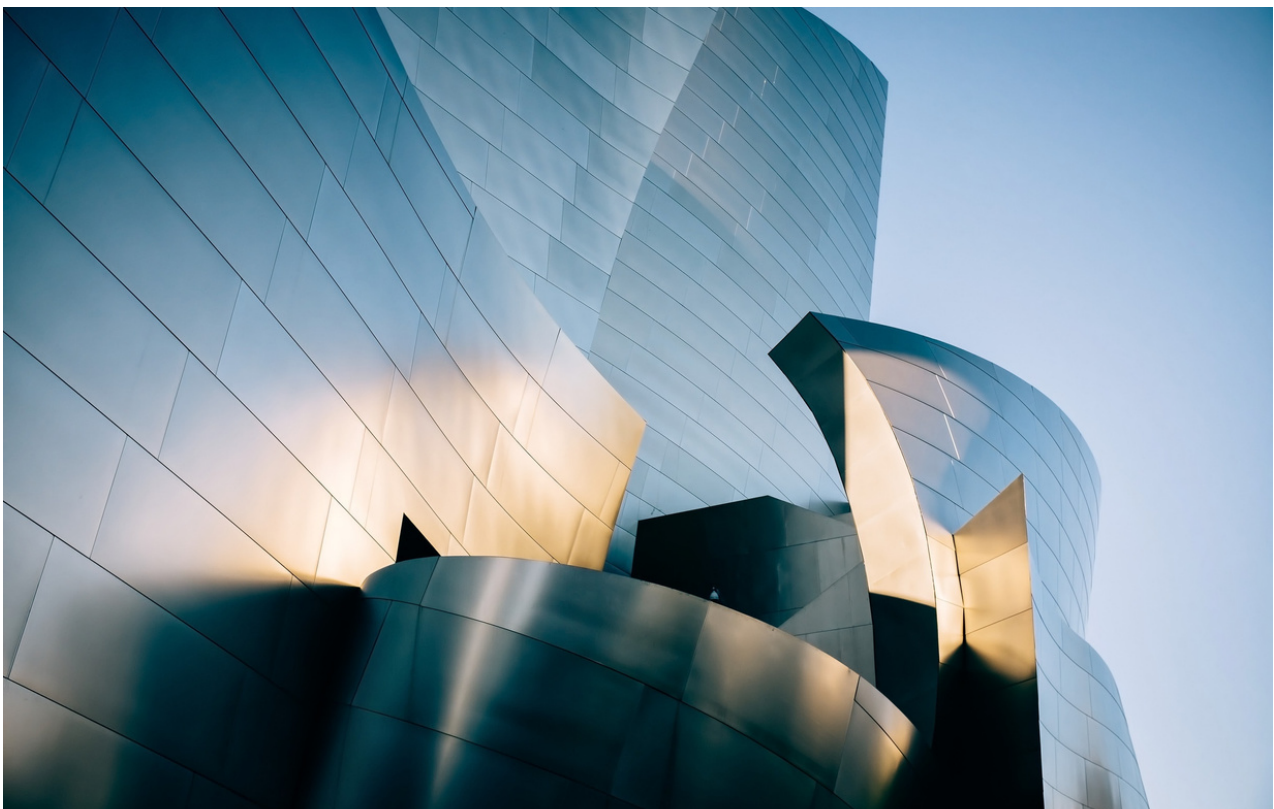
RESEARCH

For this reason, we started this research into the RPG development world. We shared a survey and we got a surprisingly high number of responses (over 100!). We learned from this experience that RPG developers are willing to share their experiences and learn from each other. Many respondents were also available to do in-depth interviews.

THE REPORT

In this report, we share what we learned from the survey and the interviews. We hope that it will be useful to anyone interested in RPG.

This has been possible only thanks to the generous participation of many RPG developers: thank you to each and every one of you!



RPG DEVELOPERS

04

YEARS OF EXPERIENCE

We got developers with experience ranging from 1 year to 50 years of experience. Given RPG was born in 1959, someone seems to have been using it from its early days.

Year of experience	% of developers
> 30 years	30%
> 20 < 30 years	42%
> 10 < 20 years	10%
< 10 years	18%

We can see how the vast majority of RPG developers (over 70%) have 20 years or more of experience. It is not surprising that RPG developers are typically experienced.

We know that attracting younger developers to RPG is a challenge, so it is encouraging to see that there is a certain number of “young” RPG developers (developers with less than 10 years of experience with RPG).

WHAT RPG DEVELOPERS LIKE IN RPG?

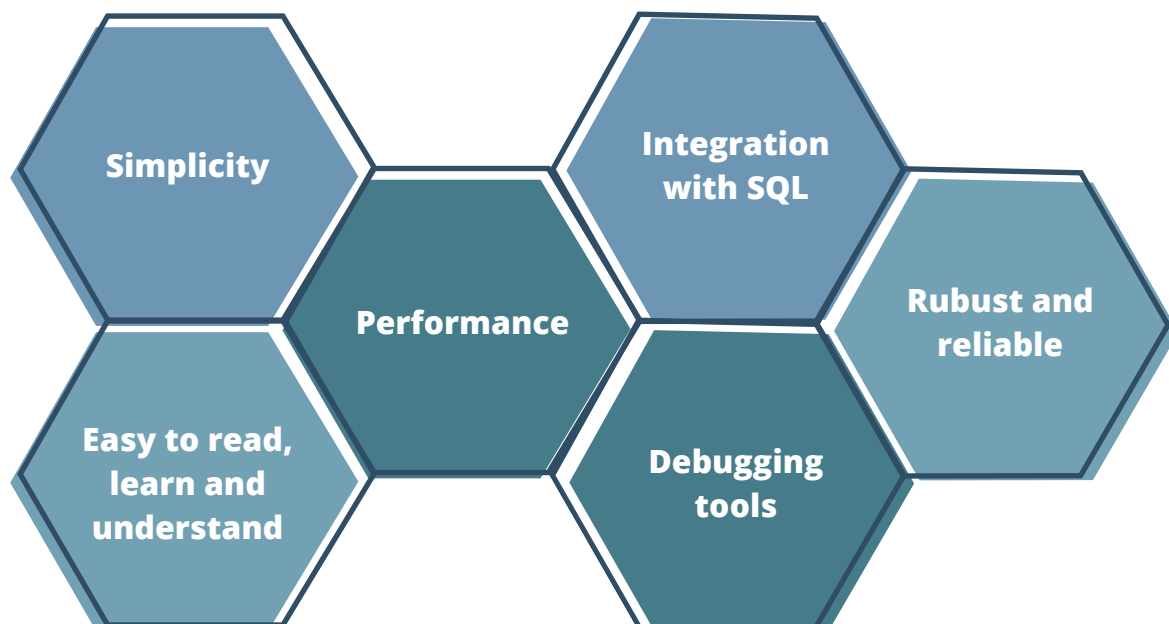
One answer came up again and again: simplicity. 37% of respondents reported this as a very important aspect and 10% indicated that the language was easy to read, learn, or understand.

They also appreciated the performance, particularly in operating with files. They also appreciated integration with SQL. One other thing they appreciated is the presence of debugging tools.

The language is perceived as robust and reliable.

Efforts from IBM to evolve the language from RPG 2 to RPG Free were appreciated. Definitely, some of the respondents seem to have experienced all these evolutions during their careers.

One answer we liked was this: “RPG lets you focus on business logic”. I think this is indicative of the philosophy behind RPG: a language that gets out of the way and lets you focus on your business.



RPG LANGUAGE

06

DIFFICULTIES IN PROGRAMMING IN RPG

The most common complaint is with the obsolescence of the programs written in RPG. This is connected to the fact that some codebases are still using old patterns and some RPG developers are not willing to move away from old ways of writing code and embrace the possibilities provided by newer versions of RPG.

One thing mentioned is the difficulty of integrating code with external systems, through web services or other means. Related to this are complaints about the fact that the IBM i system is perceived as a “closed” environment, difficult to integrate with other systems.

Some complaints contribute to the perceived lack of modern features. For example, the lack of support for unicode or integration with version control systems (like Git), also emerged as a complaint. Other difficulties are with the creation of interfaces: defining screens and UIs is not an RPG strong suit.

Smaller considerations regard the lack of functionalities to work with strings, which lead to writing more code than it would be necessary with other languages. A few respondents indicated that it is difficult to write concise code with RPG. One respondent also reported that complex mathematical expressions are difficult to express with RPG. 15% of respondents indicated that RPG have no problems at all.

Coding issues

The difficulty of integrating code with external systems, through web services or other means. Also we identify that it is difficult to write concise code with RPG.

Obsolescence

Some codebases are still using old patterns and some RPG developers are not willing to move away from old ways of writing code and embrace the possibilities provided by newer versions of RPG.

Mathematical expression

Complex mathematical expressions are difficult to express with RPG..

Funtionalities with strings

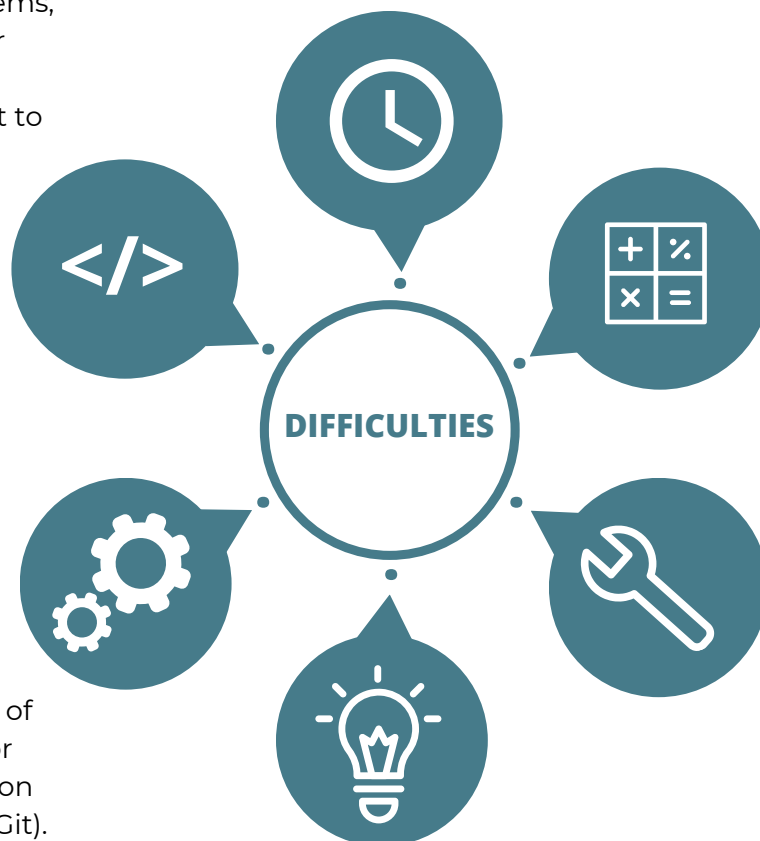
The lack of functionalities to work with strings, which lead to writing more code than it would be necessary with other languages

Lack of modern features

For example, the lack of support for unicode or integration with version control systems (like Git). Other difficulties are with the creation of interfaces: defining screens and UIs is not an RPG strong suit.

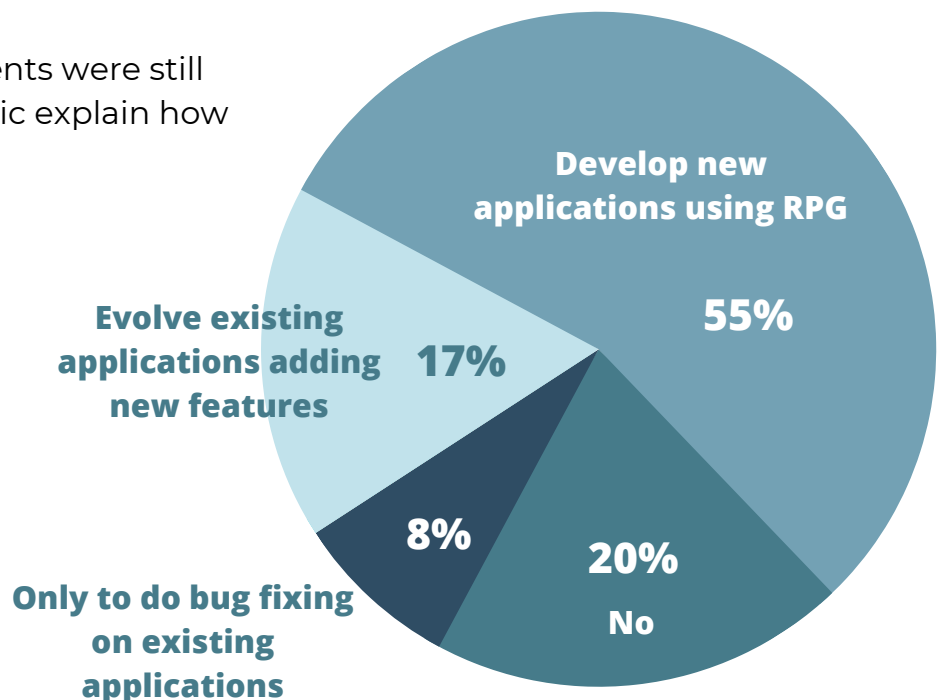
Creation of interface

Defining screens and UIs is not an RPG strong suit.



DO YOU CURRENTLY USE RPG?

We asked if respondents were still using RPG, the graphic explain how they replied.



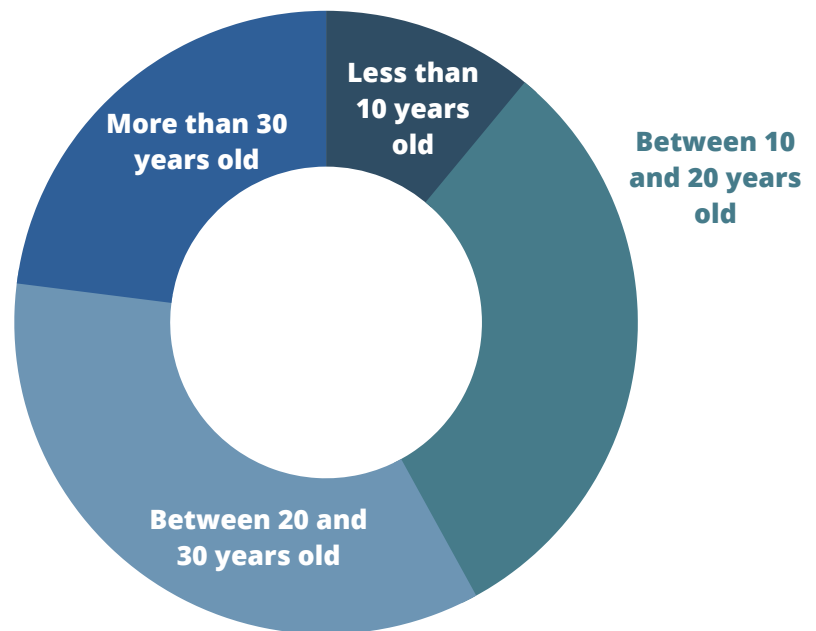
THE CODEBASE

We asked respondents how old was the codebase they were working on, in the infographic we report the results.

This trend indicates that **usage of RPG is declining as there are way more old applications than new ones.** Over half of the applications are more than 20 years old and almost 90% are more than 10 years old.

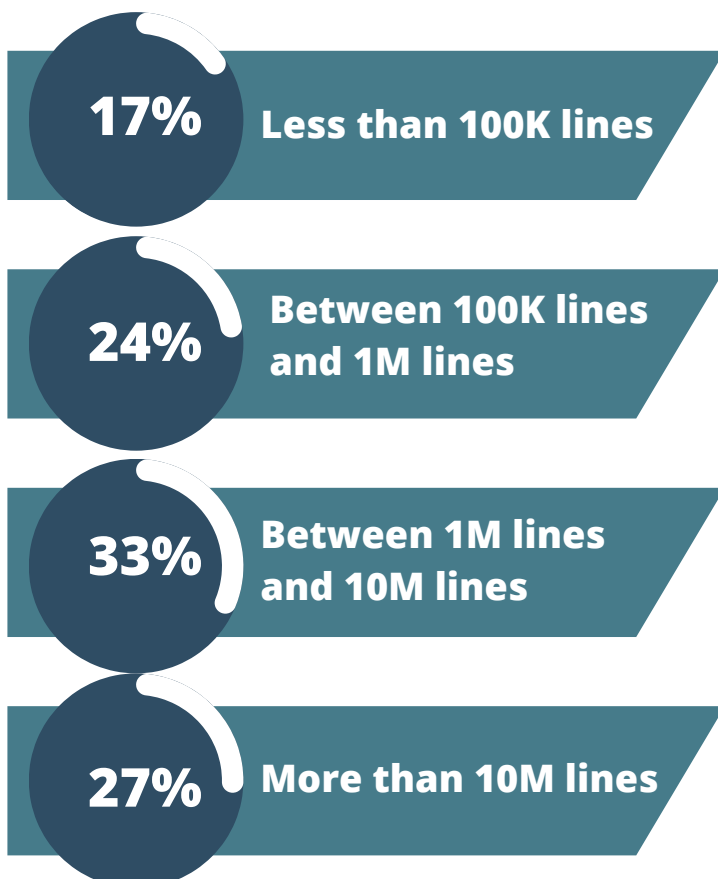
No surprises here: RPG codebases have been around for quite some time.

This of course has implications, as it means that this software was designed when the scenario was very different. Smartphones and tablets were not a thing at the time and web applications were not as widespread as they are now.



LINES OF CODE

We asked if respondents how many lines of code there were in their codebase, In the following infographic how they replied.



Codebases written in RPG are large. The majority is over 1M lines of code.

For reference, the Linux Kernel 2.2.0 and Unreal engine 3 are estimated to be around 2 million lines of code. Chrome, Firefox or the avionics & online support systems software on a Boeing 787 are estimated to be between 1 and 10 million lines of code. Microsoft Office, or any modern Windows OS are estimated to be over (and possibly a lot over 10 million lines of code).

There are factors that could make RPG codebases particularly large, like the limited structures for abstractions which encourage code repetition, but still these comparisons can give us an idea of the kind of complexity that the typical RPG developer has to deal with.

DOMAIN

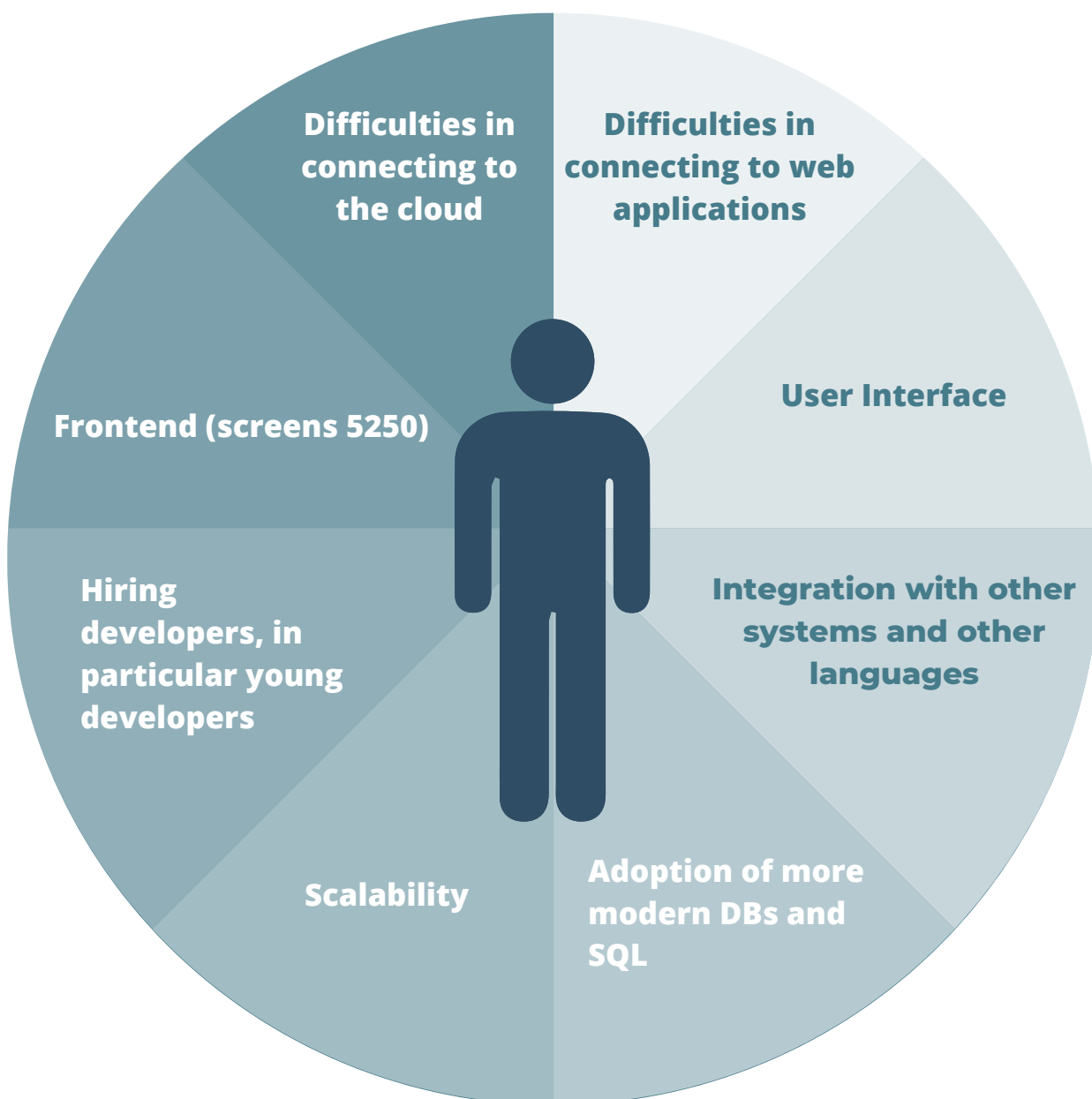
Two thirds of the software written with RPG is ERP or enterprise software: here we have clear winner. Over one fourth of the software is written for the finance domain. Also automotive seems to be another sector where RPG is used, with 3% of the software.

In our experience, we have seen that RPG is very frequently used by analysts or developers with an interest in the domain. They typically use RPG to specific customizations or organization-specific logic for larger systems. They may alternatively define formulas or strategies for financial software.

In this sense, RPG is typically used to describe valuable business logic owned by the company, which constitutes a differentiating factor and which remains valuable over time.

MAIN CHALLENGE

We asked which was the main challenge in maintaining the codebase. We got a lot of very different answers. We identify the recurring trend below.



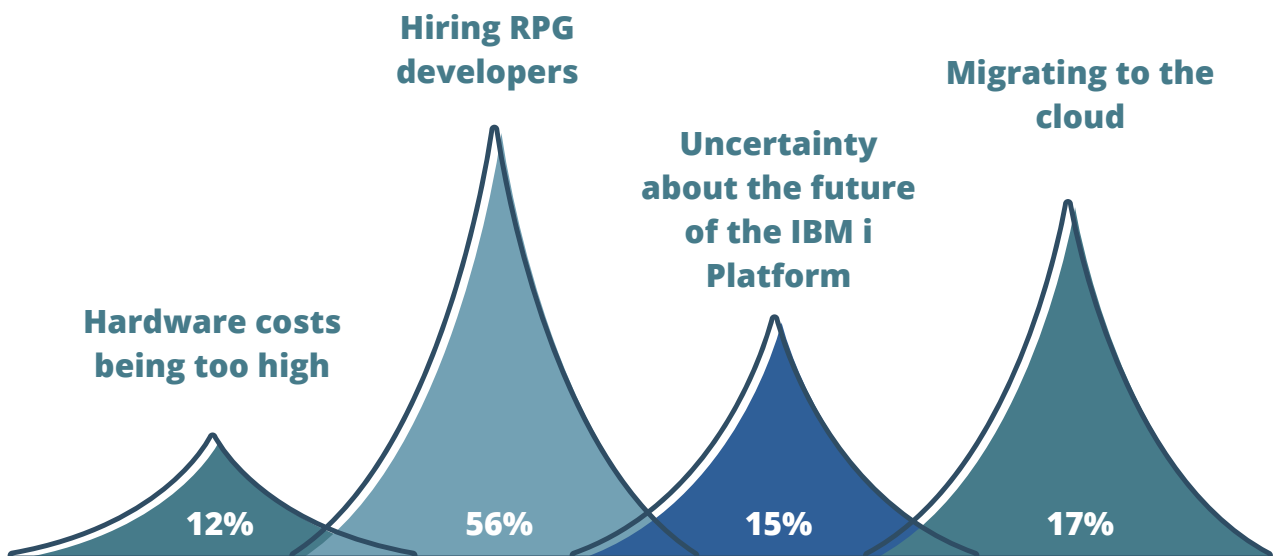
WHAT IS THE FOCUS OF THE MAIN CHALLENGES?



In general we got the feeling that the challenges are mainly focusing on the system on which the RPG code is running and not on the language itself. Compatibility with other platforms and systems is the most recurring theme.

PROBLEMS

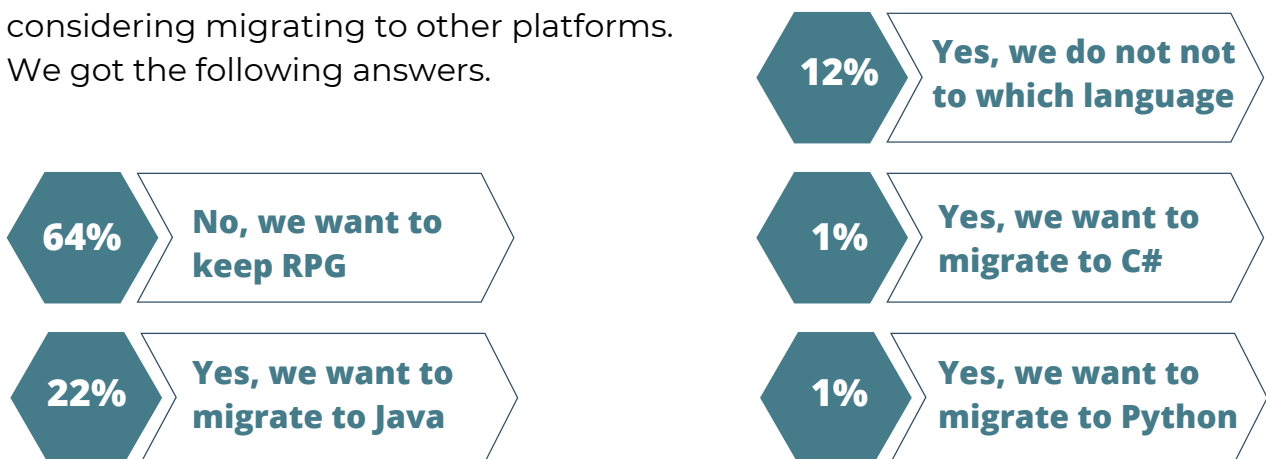
We asked the respondents which problem was the most pressing, in the list we shared. We got the following responses.



We were expecting, from anecdotal experience, that the cost of hardware was a major issue, while only one respondent out of 8 reported it. We instead underestimated the problem in hiring developers which affects over half of the respondents. Migrating to the cloud is instead an issue for close to one fifth of the respondents, and it would be interesting to observe the trend over the years in the number of companies looking to migrate to the cloud.

MIGRATION

We asked the participants if they were considering migrating to other platforms. We got the following answers.



It should be noted that we asked developers, who are mostly RPG developers and who may have an interest in keeping RPG codebases. It would be interesting to compare these answers with the answers provided by CTOs or IT managers.

While we were aware that most companies are considering migrating to Java, this survey seems to indicate that Java is practically the only target language considered, with just a tiny minority considering Python. It would be interesting to understand the reasons why Java is winning so overwhelmingly over other valid candidates. From the number of respondents who have not decided which language to migrate, it would be interesting to learn more about the factors they are taking into account, and which possible target languages they are considering.

ECOSYSTEM

From analyzing the answers to other questions we could make some considerations on the ecosystem around RPG programs.

There were not many responses citing tools or libraries, and we think this is an indication that the ecosystem is not providing many solutions.

Both the lack of libraries and of tools was reported explicitly as an issue.

Tools there were cited were IBM Rational Developer for i and X-Analysis.

OTHER

We finally asked if there was anything else the participants wanted to share. We got a lot of very different and interesting responses. Here we would like to share a few:

- Many respondents indicated satisfaction with the evolution of the language over the years and the solidity of the platform
- Several respondents highlight the need to create a community like is the case for Java and Ruby. Similarly they shared the need to better promote RPG and attract younger developers
- There are recurring doubts about IBM's plans for the platform
- Someone cited the desire to have RPG running also on other platforms
- Some observed a trend in migrating from RPG to SAP
- There are contrasting opinions about the free format: some love it, while others prefer the fixed format



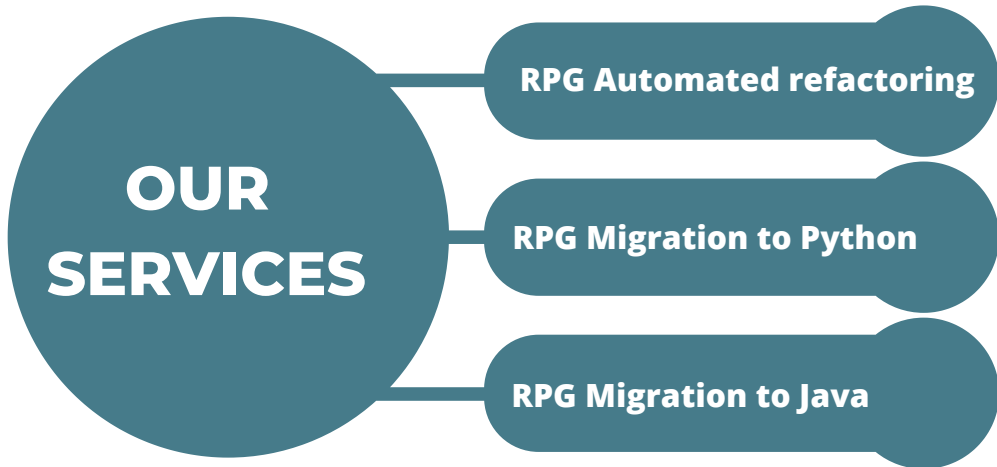
CAN YOU BETTER LEVERAGE THE COMPETENCIES YOU HAVE, AND THE PROCESSES YOU DEVELOPED BY USING MORE APPROPRIATE LANGUAGE AND TOOLS?

In this era the main asset most companies have is knowledge.


We produce knowledge, verify it, analyze it, elaborate it, put it into action. We do that by means of languages and tools.

The services we offer aim to put you in the position of getting the most value from the people you have. Professionals use the best tools and the most important tools used by knowledge workers are languages.

We have a range of services dedicated to the RPG Language to help our customer to achieve their goals.



THANK YOU



We hope you found this report interesting and we would love to hear your feedback. Please let us know if you found some results surprising, if something is missing or if you would like us to correct anything.

The project development is focused on the Italian industry, and we are planning to develop the same report for other regions.



info@strumenta.com



www.strumenta.com

