



HOW TO TAKE REST WITH LEGACY SYSTEMS



BUSINESS PROBLEM

For more than 30 years, IBM i server have earned the well-deserved fame of reliable “workhorse”. High performance, reasonable cost and absolute reliability are recognized by all customers who use this server. It is a reliable fortress for their data and processes. However, in the modern world of open systems, strong walls of the fortress are more likely to impede development. More and more companies – manufactures, banks, IT companies, insurance, travel agencies - are dealing with an ultimate need to modernize their existing legacy systems to speed up digital transformation and safely exchange data with their partners - all of this to improve user expertise, increase conversion and introduce competitive solutions and services to their clients.

The main problems faced by owners of IBM i:

- Implementation of new integration standards is a complex and expensive process. This is partially due to the long history of the platform itself and the volume of accumulated expertise, data and processes
- Slow implementation of new technologies and standards in the native operating system

In order to keep up with the modern needs of IT industry, these companies typically start long, difficult and expensive modernization projects – buy new hardware and software and even, sometimes, move from IBM i to another platforms.

They all need a simple and effective way to transform their Legacy systems into an open, friendly world of modern service-oriented solutions.



THE PROBLEM SCALE

IBM I Marketing Survey 2020

The market for IBM i continues to be extremely stable with 1.4% annual attrition.

Although we've grown accustomed to hearing that the IBM i market is shrinking, its fan base continues to expand. 65% of IBM i shops are planning to expand their IBM i footprint or keep it as is for 2020, which significantly outnumbers those who plan to migrate off the platform. Also, 72% rely on IBM i to run more than half of their core business applications, a strong indication of the platform's stability.

Despite large number of companies are already in the process of digital transformation, the problem of legacy modernization still represents one of the top concerns in the respective market segment

What are your top concerns as you plan your IT environment?





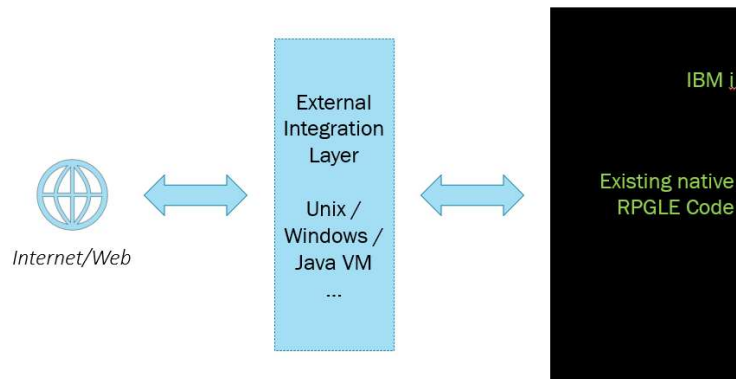
GENERAL APPROACH

There is no universal approach to modernization. As a rule, each API has to be modified separately, and overall modernization efforts are growing with increasing number of API

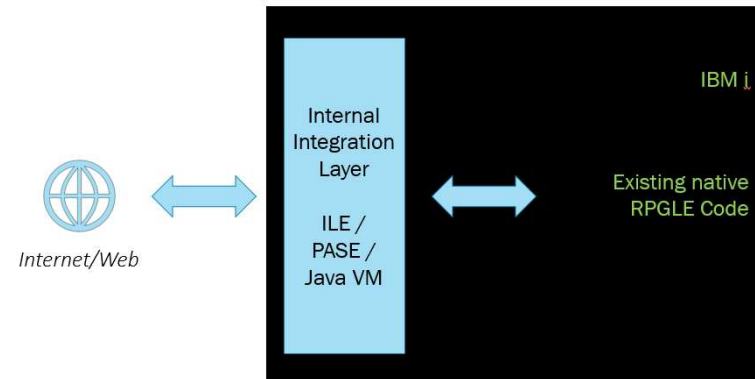
Typical Legacy system on IBM i has hundreds if not thousands existing RPGLE APIs and tables, so as a rule, typical modernization project is a long, complex and expensive project.

There are 2 options, that are usually considered

Option A – External Integration Layer



Option B – Native Integration Layer





KEY CHALLENGES

Both approaches require **significant amount of custom code** for

- Input/output parameters transformation
- Messages delivery
- Authority verification
- Credentials mapping
- Business or technological process transformation

Keep in mind

- **Performance and availability.** Option A adds extra links to the processing chain, regarding option B - Java and Unix (PASE) applications, although they work on this platform, can not be compared with native implementations in terms of speed and resource consumption
- **Development and support cost.** Native IBM i technology skills, such as development in the RPG language, are increasingly rare



EXISTING SOLUTIONS ON A MARKET

| READY TO USE SERVERS | DEVELOPMENT LIBRARIES | OTHER |
|--|--|--|
| IWS - Integrated Web Services for IBM i Node.js for IBM i | HTTPAPI IBM i system API RPG-XML Suite | Rest4i Mulesoft AS400 connector iWebSrv |
| <p>PROS</p> <ul style="list-style-type: none">FreeEasy to installRelatively easy to find developers <p>CONS</p> <ul style="list-style-type: none">Java/PASE based – high hardware resource consumptionDifficult integration with native applicationsThere is no easy solution to transform existing software | <p>PROS</p> <ul style="list-style-type: none">Help developers upgrade existing code with less efforts <p>CONS</p> <ul style="list-style-type: none">Requires server-side developmentStill requires to upgrade existing code | <p>PROS</p> <ul style="list-style-type: none">A set of automation tools, methodologies and architecture solutions for the modernization process <p>CONS</p> <ul style="list-style-type: none">Different for different products – architecture, performance, usability etc. |

WHAT IS I2REST BUSINESS VALUE



- Easy to use solution allowing to expose your existing legacy RPGLE programs as Open APIs and serve them - **in minutes, Open your APIs to your customers and let them participate more closely in your business processes**
- **It fully eliminate RPGLE development efforts** - you can use leave your legacy programs AS IS. Typical **implementation cycle reduces from months to several days**
- With supplied i2Rest **OAuth2 client**, your IBM i applications can get access and call countless Open APIs of other suppliers – Google, IBM, Twitter, Facebook, PayPal ... Enhance your business processes with seamless integration of your IBM i to Google Drive/Mail/Print, PayPal payments etc.
- Your existing IBM i machine becomes **fully functional OAuth2 server and client**, new hardware or software is not required
- **Attractive Licensing Model.** Even with Free edition, you can use many valuable features of a product

WHAT IS I2REST SERVER



- Ready to use IBM i native REST JSON Web Server
 - Any non-interactive RPGLE program can become fully functional OpenAPI in a minutes
 - Automatic creation of OpenAPI 3.0 API description files
 - Native OAuth2 server implementation
 - Authorization code/Device/Client credentials/Refresh token flows
 - Pluggable security model
- Simple, performant and lightweight (black box – interaction only via REST)
 - No Java, No PASE
 - Native IBM i implementation
 - Security/ authentication
 - Hardware SSL protection
 - Native calls for existing RPGLE code



WHAT IS I2REST TRANSFORMATION PROCESS



1

- Install and start i2Rest server on IBM i
- Prepare PCML descriptions for your RPGLE programs
- Publish PCML to i2Rest

```
Dalme ... 1 121 Edit
i2Rest>
***** Beginning of data *****
0001.00 <pcml version="1.0">
0002.00   <program name="echo" path="/QSYS.LIB/WLIB.LIB/NTPGM.PGM">
0003.00     <data name="echo" usage="inputoutput" type="char" length="10" trim="both"/>
0004.00   </program>
0005.00 </pcml>
***** End of data *****
```

2

- Your RPGLE is accessible as OpenAPI/ OAuth2 Rest JSON service
- Try it in Postman/Swagger/SoapUI etc.
- Use it with any client library



WHAT IS I2REST SERVER

KEY DIFFERENTIATORS



| PARAMETERS | I2REST SERVER | IBM IWS |
|--|---|---|
| ARCHITECTURE & IMPLEMENTATION DETAILS | | |
| Server Engine | Native implementation of high performant multithreaded HTTP web server (based on gSOAP toolkit) | IBM HTTP Server for IBM i, based on Apache Tomcat |
| Development language | IBM ILE C | Java |
| Open API specification | 2.0, 3.0 | 2.0 |
| User authentication | OAuth2 (Bearer), HTTP Basic, SSL certificate | HTTP Basic |
| User authorization | Provided by OS + OAuth2 or custom authorization model | Provided by OS |
| Supported OAuth2 flows | Authorization code, Client credentials, Refresh token, Device code | None |
| PERFORMANCE | | |
| TPS (transactions per second) | 1100 | 65 |
| Memory consumption | 29mb, 31 mb in peak | 200 mb, 371mb in peak |

WHAT IS I2REST OAUTH2 CLIENT FOR IBM i

- Client commands:
 - Request OAuth2 device authorization
 - Query OAuth2 authorization status
 - Send/receive API requests with OAuth2 authentication
- Client API library for use in RPGLE



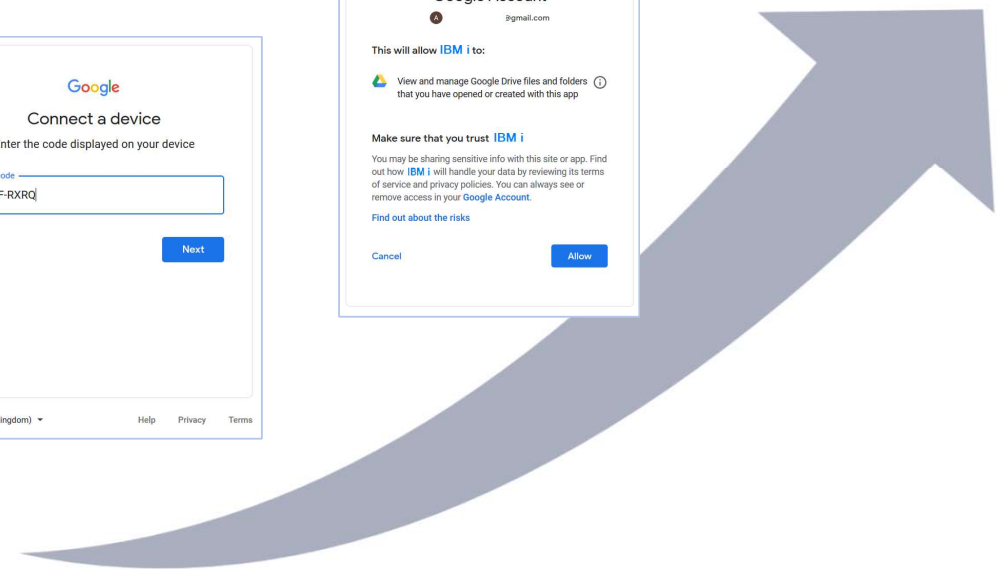
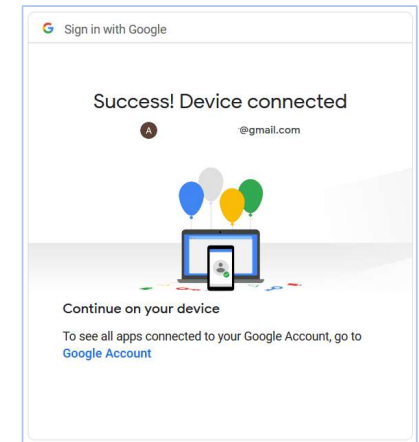
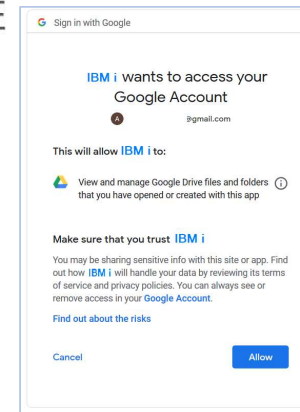
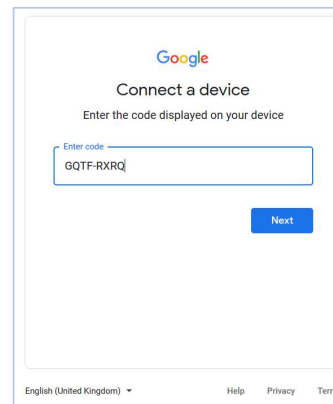
```
Request needs OAuth2 authorization
Please open this URL in your browser
https://www.google.com/device
and enter code shown below
GQTF-RXRQ

Or enter user name to request authorization
_____

Pending (1733 seconds remained)...

F5=Query authorization status, F12=Cancel
```

18/060



WHAT IS I2REST CLIENT

KEY DIFFERENTIATORS



| PARAMETERS | I2REST CLIENT | BVS Tools GETURI |
|---|--|--|
| ARCHITECTURE & IMPLEMENTATION DETAILS | | |
| Authentication / authorization | HTTP Basic/ user-defined HTTP headers, OAuth2 bearer token | HTTP Basic/ user-defined HTTP headers |
| File uploads | 0 - Many | 0-1 |
| Authorization flow, device flow, client credentials flow | OAuth2 conformance | None |
| Supported OAuth2 flows | Authorization code, Client credentials, Refresh token, Device code | None |
| PRICE | | |
| Price | Free | 0,99\$ per day |
| Use Cases | Free Examples | 0,99\$ per day per client of each type (eg Google API, Microsoft, PayPal) |

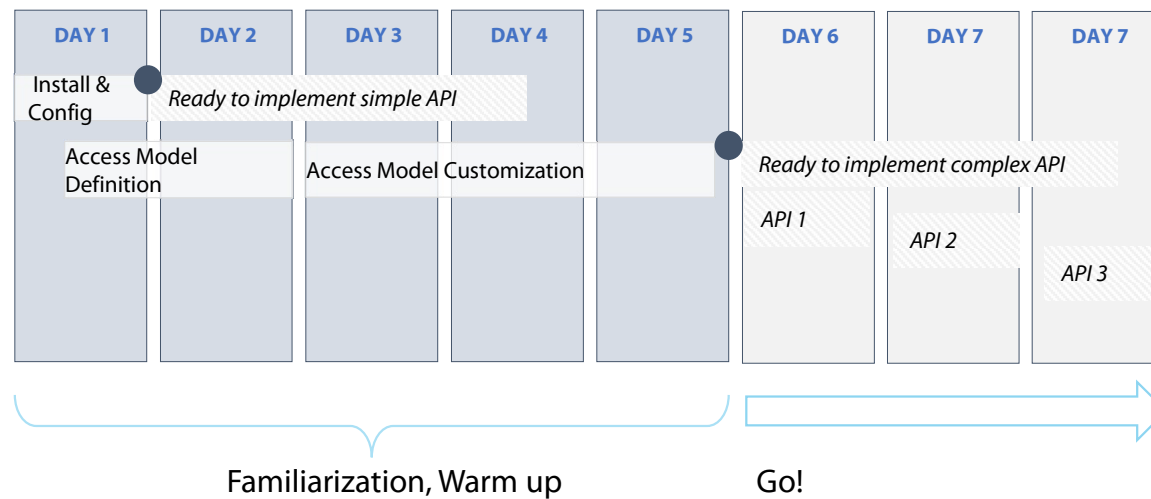
WHAT IS I2REST

TYPICAL IMPLEMENTATION TIMELINE



i2Rest requires a minimum time to start serving/consuming Open API
You can start using i2Rest Server and Client almost immediately after installation

If you need a more complex security model, add several days for planning, development and implementation



WHAT IS I2REST LICENSE MODEL



| FEATURE(S) | FREE EDITION | PREMIUM EDITION |
|---|------------------|------------------|
| I2REST CLIENT | | |
| Call OAuth2 APIs | ✓ | ✓ |
| Execute http(s) client requests (get, post, put, del) | ✓ | ✓ |
| i2Rest Client API to enhance existing RPGLE programs | ✓ | ✓ |
| I2REST SERVER | | |
| Built-in Authorization Model | ✓ | ✓ |
| Ability to extend authorization model | ✗ | ✓ |
| #APIs per i2Rest server instance | Up to 10 APIs | Unlimited |
| #OAuth2 Users per i2Rest server instance | Up to 5 users | Unlimited |
| #OAuth2 Clients per i2Rest server instance | Up to 5 clients | Unlimited |
| #OAuth2 Scopes per i2Rest server instance | Up to 5 scopes | Unlimited |
| #API calls | Unlimited | Unlimited |
| Serve local and/or remote RPGLE as API | Local and Remote | Local and Remote |

WHAT IS I2REST

WHATS NEXT



Need more details?

Please visit www.i2rest.com

or email to contacts@i2rest.com