Integrating RPA with App Connect

23rd September 2021

Nigel T. Crowther

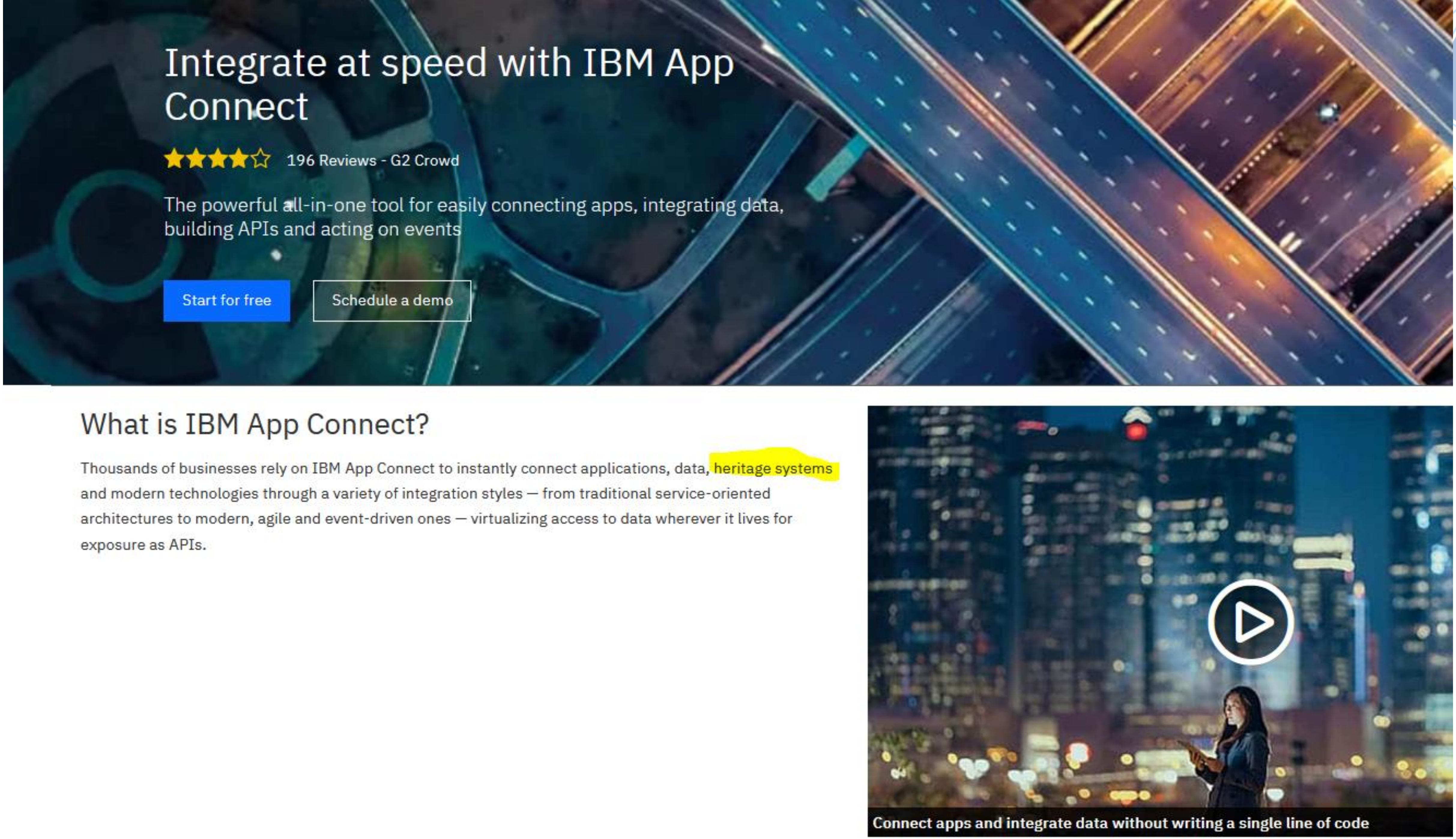
RPA Technical Sales - EMEA

Digital Business Automation

ncrowther@uk.ibm.com +44 7480794980

outthink limits

#IBMCloud



Integrate at speed with IBM App Connect

★★★★☆ 196 Reviews - G2 Crowd

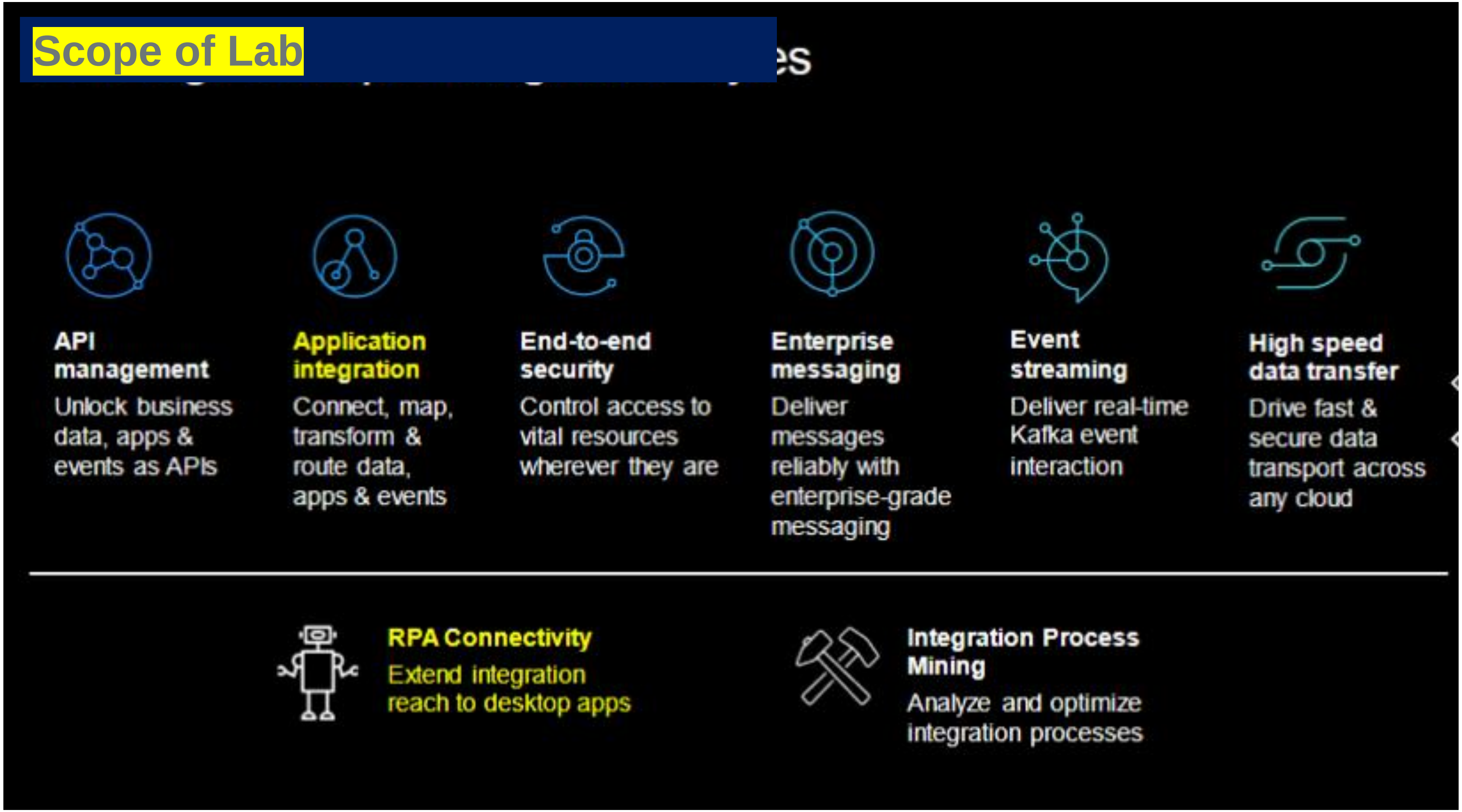
The powerful all-in-one tool for easily connecting apps, integrating data, building APIs and acting on events

What is IBM App Connect?

Thousands of businesses rely on IBM App Connect to instantly connect applications, data, heritage systems and modern technologies through a variety of integration styles — from traditional service-oriented architectures to modern, agile and event-driven ones — virtualizing access to data wherever it lives for exposure as APIs.

Connect apps and integrate data without writing a single line of code

Where RPA sits in CP4I



Scope of Lab

es

API management
Unlock business data, apps & events as APIs

Application integration
Connect, map, transform & route data, apps & events

End-to-end security
Control access to vital resources wherever they are

Enterprise messaging
Deliver messages reliably with enterprise-grade messaging

Event streaming
Deliver real-time Kafka event interaction

High speed data transfer
Drive fast & secure data transport across any cloud

RPA Connectivity
Extend integration reach to desktop apps

Integration Process Mining
Analyze and optimize integration processes

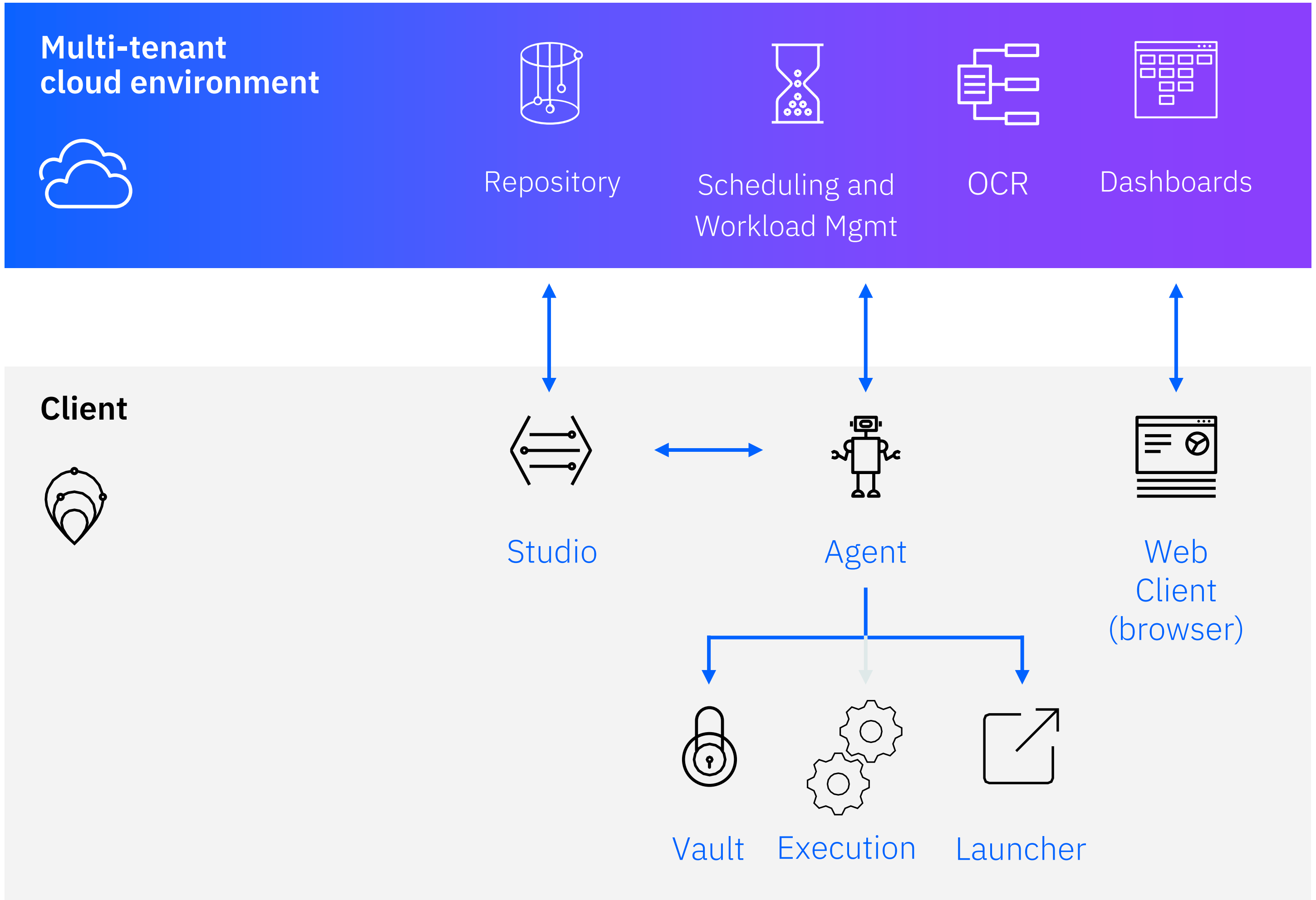
What is RPA?

RPA accesses the parts other software cannot reach!

An RPA Bot is just a script that bypasses APIs and goes in via the user interface

The bot script is fronted with a API.

High level architecture:



IBM RPA

Product

Capabilities

Unattended/Attended bots

- Unattended - automate repetitive tasks without human intervention.
- Attended - Enables human workforce to augment work using bots

AI Capabilities

- Drag and Drop AI Commands for extraction, machine learning implementation, and applied knowledgebase AI
- Train Machine Learning engines in a straightforward native manner within Studio

Intelligent virtual agent (IVAs) chatbots

- Combine chat and RPA commands to create chatbots through multiple channels that can provide engaging client interactions.

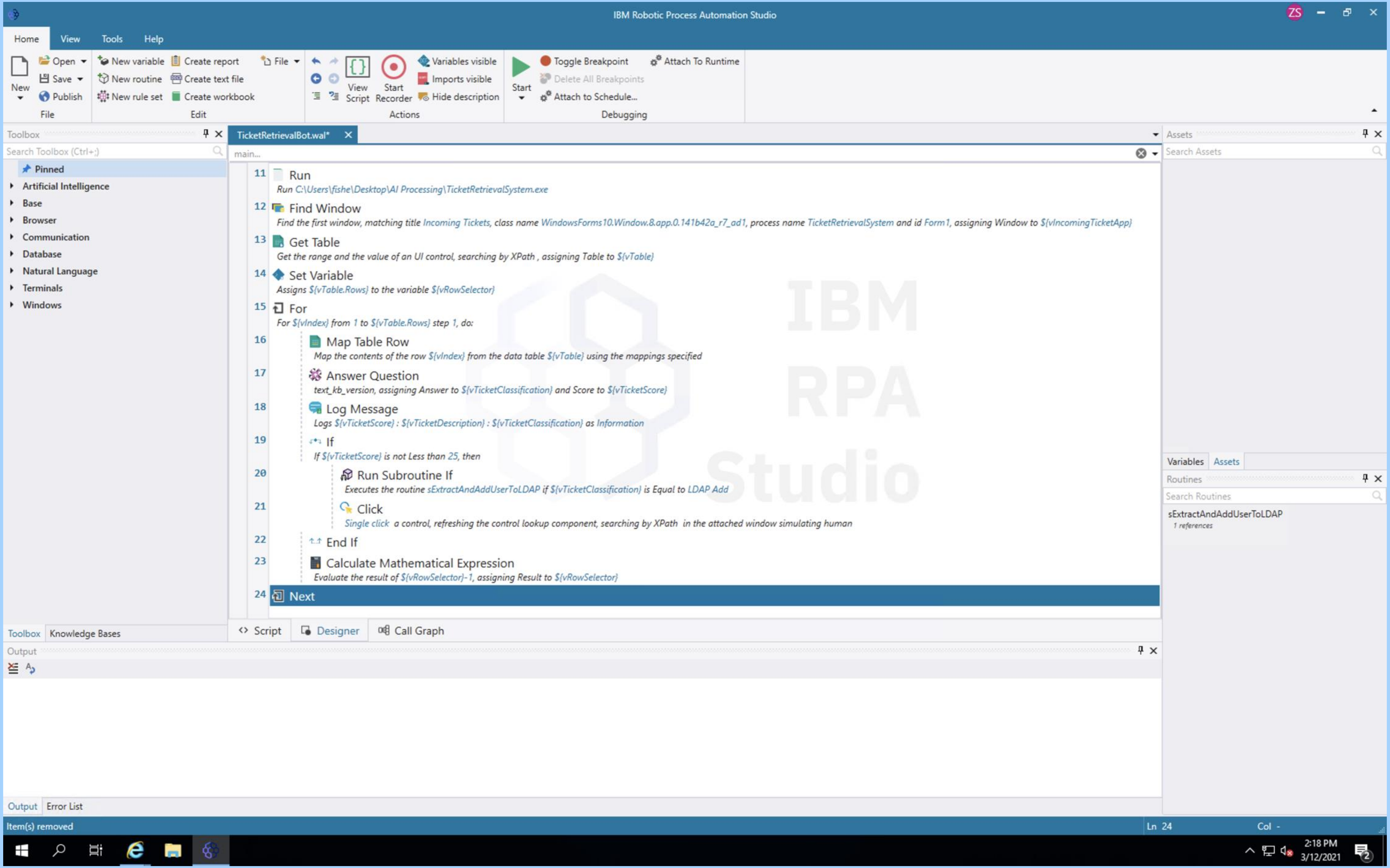
Concurrent Execution

- Allow for multiple bots to run on the same machine at the exact same time.

Dashboards

- Gain business insights into business operations.
- Prebuild the dashboards with included easy-to-use creator

IBM RPA
Studio



```
11 Run
   Run C:\Users\fishe\Desktop\AI Processing\TicketRetrievalSystem.exe
12 Find Window
   Find the first window, matching title Incoming Tickets, class name WindowsForms10.Window.8.app.0.141b42a_r7_ad1, process name TicketRetrievalSystem and id Form1, assigning Window to ${vIncomingTicketApp}
13 Get Table
   Get the range and the value of an UI control, searching by XPath , assigning Table to ${vTable}
14 Set Variable
   Assigns ${vTable.Rows} to the variable ${vRowSelector}
15 For
   For ${vIndex} from 1 to ${vTable.Rows} step 1, do:
16     Map Table Row
       Map the contents of the row ${vIndex} from the data table ${vTable} using the mappings specified
17     Answer Question
       text_kb_version, assigning Answer to ${vTicketClassification} and Score to ${vTicketScore}
18     Log Message
       Logs ${vTicketScore} : ${vTicketDescription} : ${vTicketClassification} as Information
19     If
       If ${vTicketScore} is not Less than 25, then
20         Run Subroutine If
           Executes the routine sExtractAndAddUserToLDAP if ${vTicketClassification} is Equal to LDAP Add
21         Click
           Single click  a control, refreshing the control lookup component, searching by XPath  in the attached window simulating human
22     End If
23     Calculate Mathematical Expression
       Evaluate the result of ${vRowSelector}-1, assigning Result to ${vRowSelector}
24 Next
```

IBM RPA Differentiators

01

Ease of Use

Low-code enables business users to build bots

02

Integration Points

Integrates with most IBM Cloud Paks, including CP4I.

03

AI Processing

Native NLP and easy to use interface to build an exposable chatbot for both internal and external usage

04

Cost Savings & Scalability

Concurrent Execution allows for ”doing more with less”. Grow without infrastructure.

FOR IBM INTERNAL AND BUSINESS PARTNER USE ONLY

Why RPA with App Connect?

App Connect connects systems.

BUT what about legacy systems?

RPA bypasses APIs and goes in via the standard user interface.

App Connect can call RPA via a standard API

App Connect and RPA Opportunities

Customer has App Connect and integration point with no API

Use RPA with App Connect if you need to connect to systems without APIs

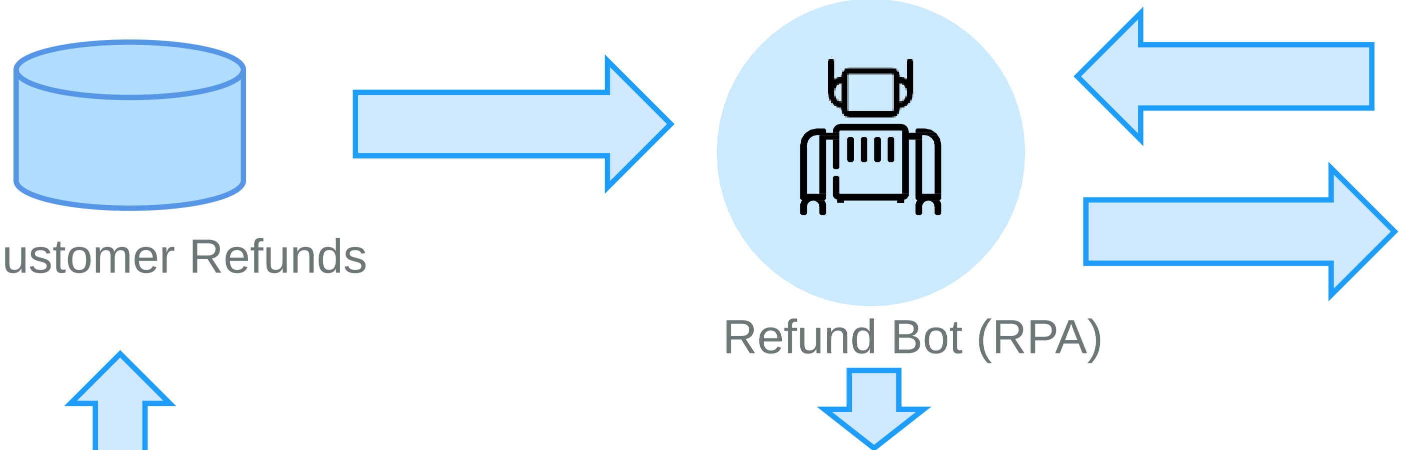
Customer has RPA and complex integration

Use App Connect with RPA if you have complex integration requirements

Real Example: Customer Refunds

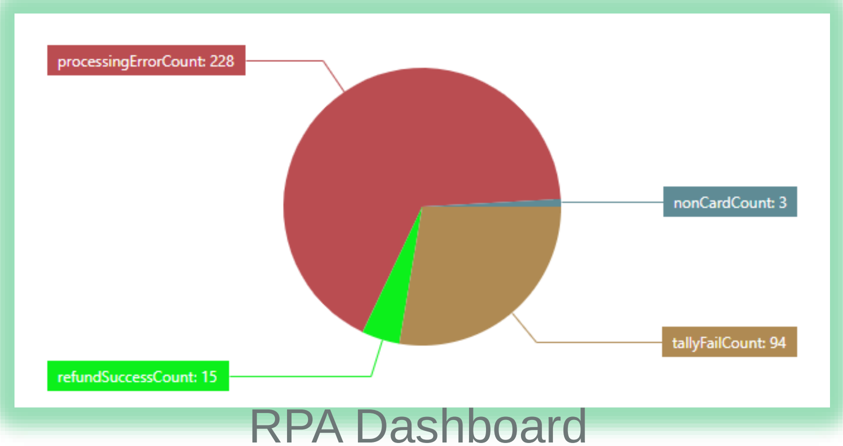
- Travel company affected by Covid
- Overloaded issuing manual refunds
- Refunds could not be automated through API
- Solution was RPA
- This use case combines IBM RPA and App Connect

Real Example: Customer Refunds

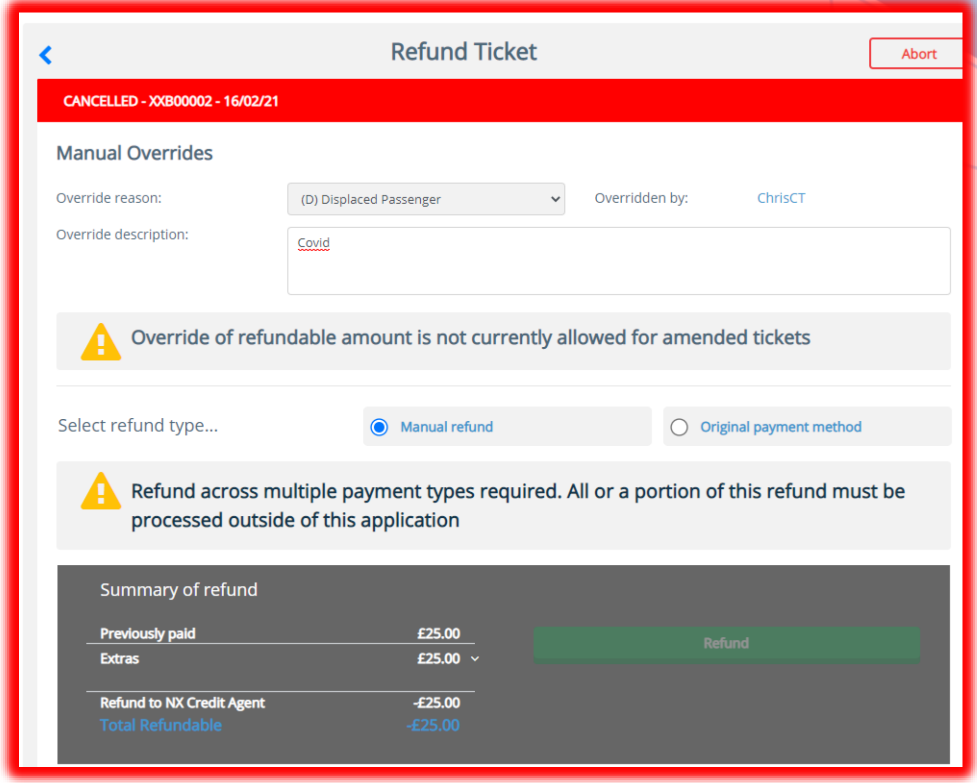


SLACK
Start Refunds

SLACK
SUCCESS!



RPA Dashboard



CANCELLED - XXB00002 - 16/02/21

Manual Overrides

Override reason: (D) Displaced Passenger

Overridden by: ChrisCT

Override description: Covid

⚠ Override of refundable amount is not currently allowed for amended tickets

Select refund type... ● Manual refund ○ Original payment method

⚠ Refund across multiple payment types required. All or a portion of this refund must be processed outside of this application

Summary of refund

Previously paid	£25.00
Extras	£25.00
Refund to NX Credit Agent	-£25.00
Total Refundable	-£25.00

Refund

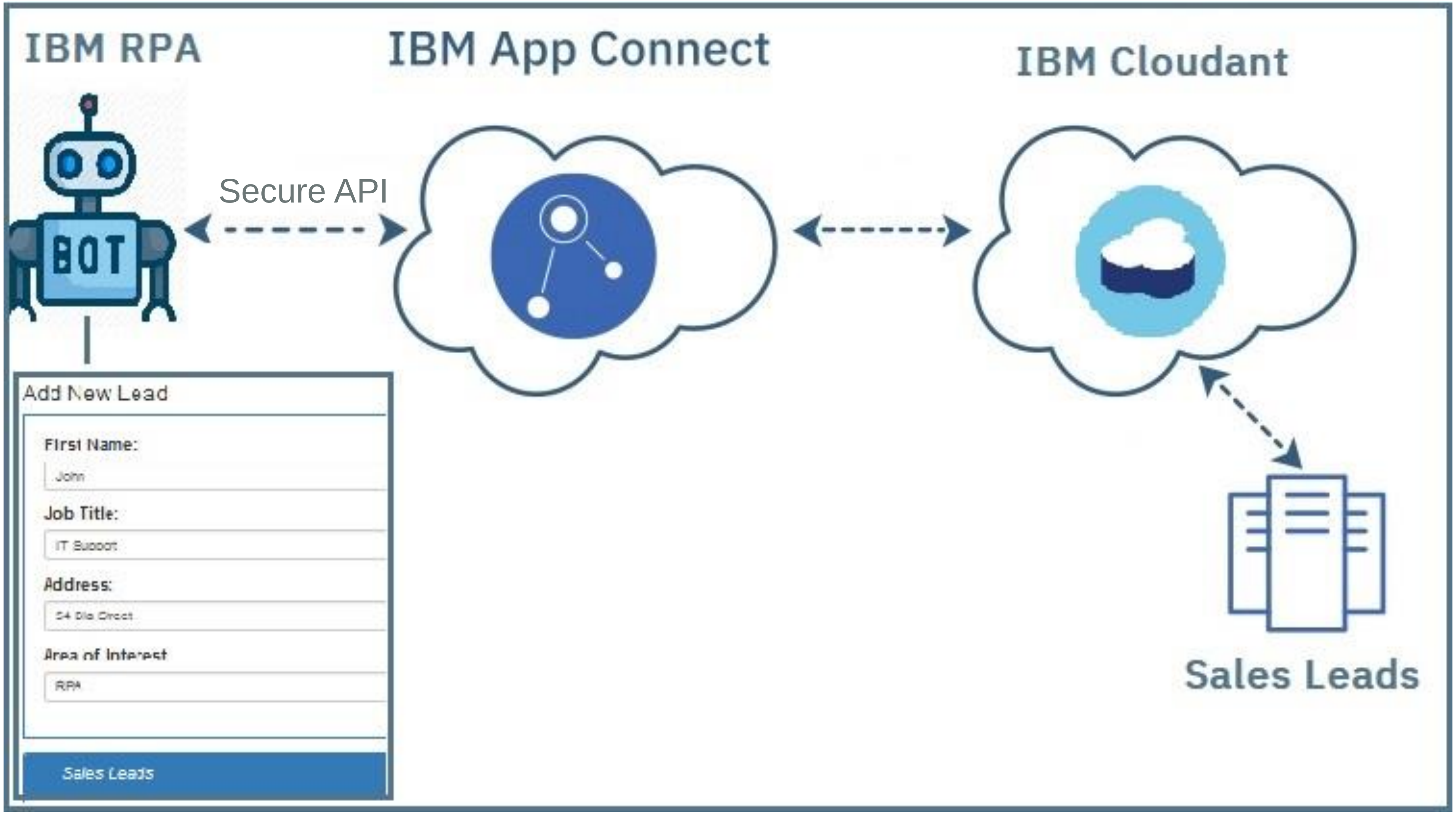
Web Site

https://youtu.be/_BL6w0bZlJ8

Demo

FocusCorpRefunds

Lab Solution Architecture



Add New Lead

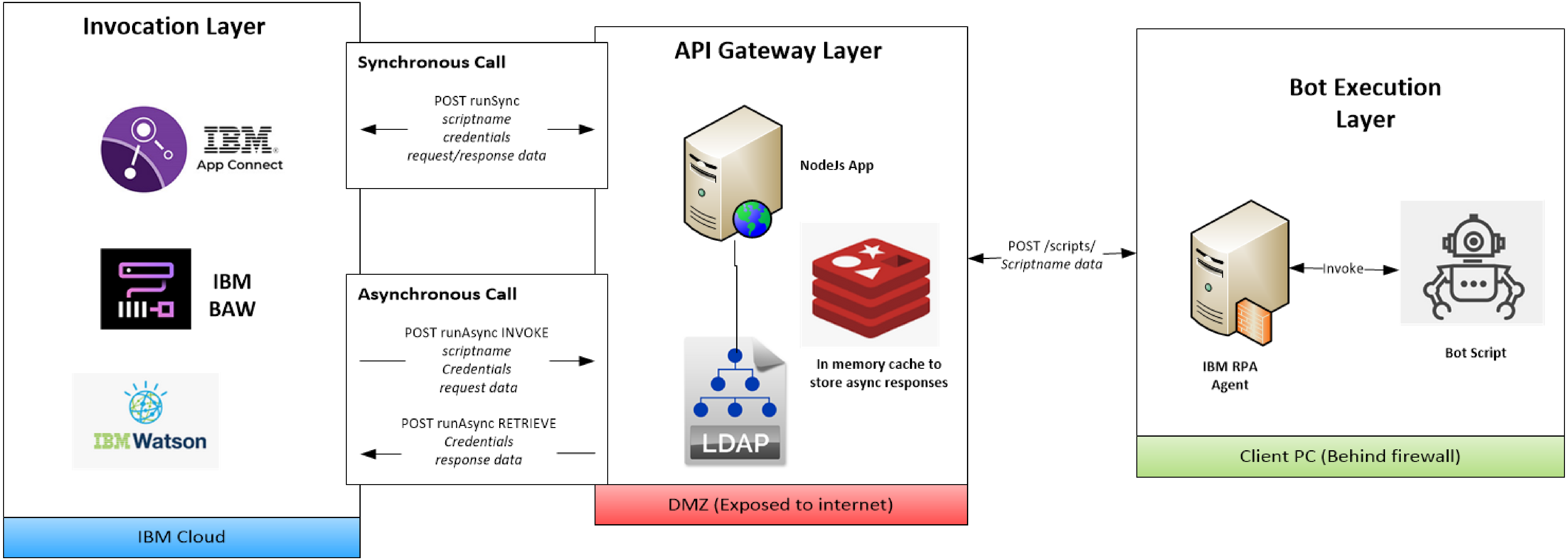
First Name:
John

Job Title:
IT Support

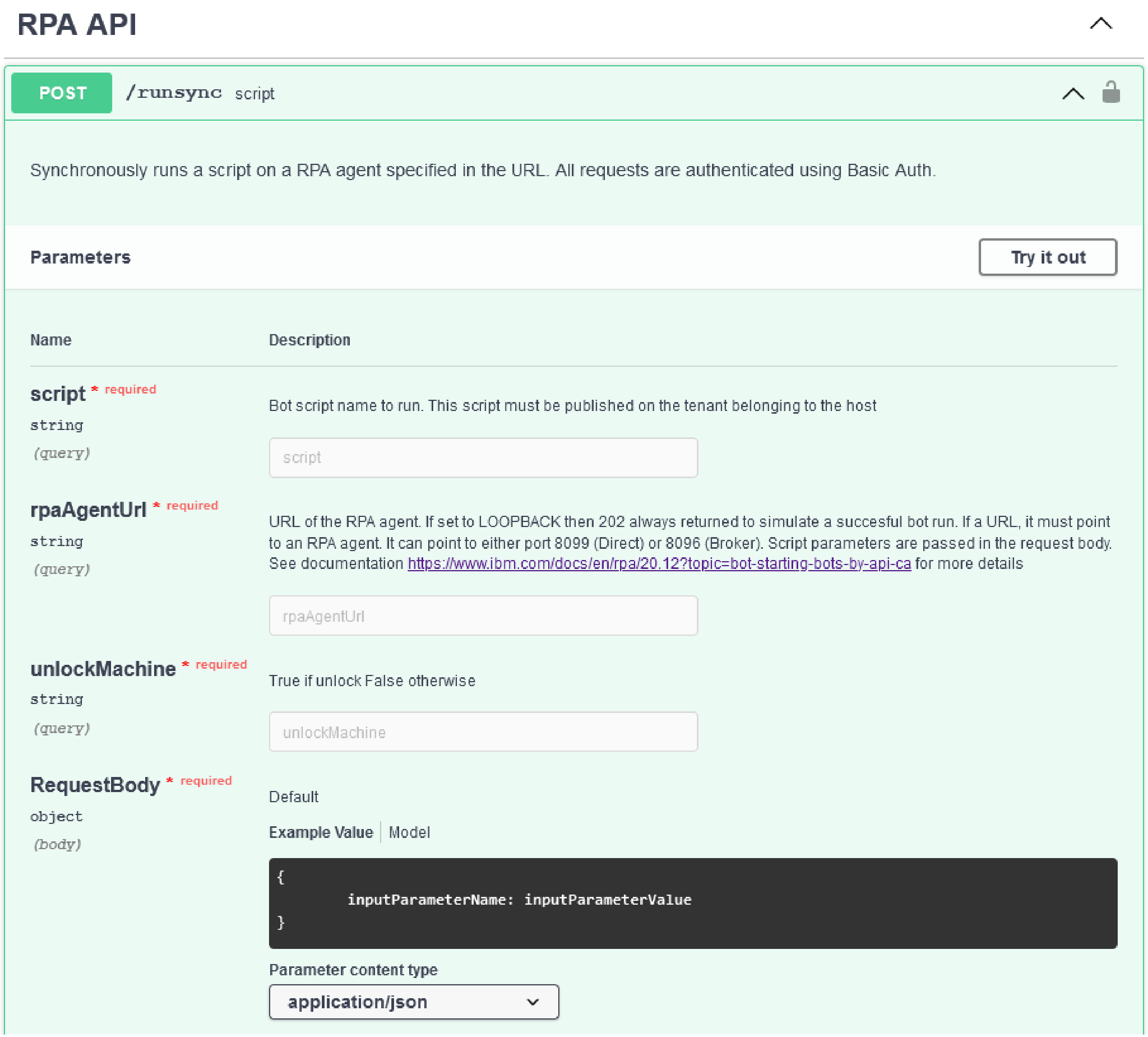
Address:
54 Ole Street

Area of Interest
RPA

RPA API Architecture



RPA Synchronous API



RPA API

POST /runsync script

Synchronously runs a script on a RPA agent specified in the URL. All requests are authenticated using Basic Auth.

Parameters

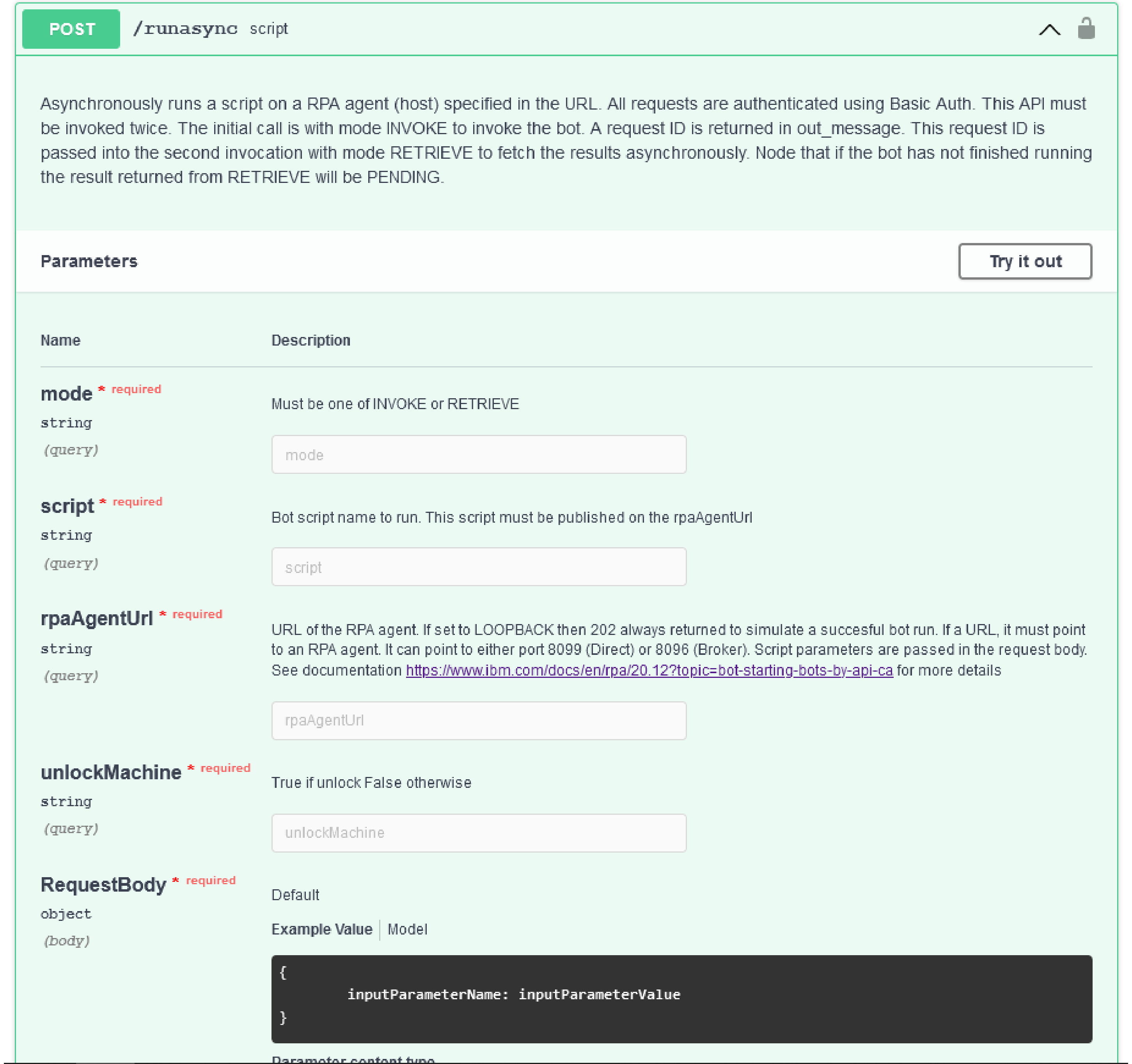
Name	Description
script * required string *(query)*	Bot script name to run. This script must be published on the tenant belonging to the host script
rpaAgentUrl * required string *(query)*	URL of the RPA agent. If set to LOOPBACK then 202 always returned to simulate a succesful bot run. If a URL, it must point to an RPA agent. It can point to either port 8099 (Direct) or 8096 (Broker). Script parameters are passed in the request body. See documentation https://www.ibm.com/docs/en/rpa/20.12?topic=bot-starting-bots-by-api-ca for more details rpaAgentUrl
unlockMachine * required string *(query)*	True if unlock False otherwise unlockMachine
RequestBody * required object *(body)*	Default **Example Value** | Model

```
{
        inputParameterName: inputParameterValue
}
```

Parameter content type

application/json

RPA Asynchronous API



POST `/runasync` script

Asynchronously runs a script on a RPA agent (host) specified in the URL. All requests are authenticated using Basic Auth. This API must be invoked twice. The initial call is with mode INVOKE to invoke the bot. A request ID is returned in out_message. This request ID is passed into the second invocation with mode RETRIEVE to fetch the results asynchronously. Node that if the bot has not finished running the result returned from RETRIEVE will be PENDING.

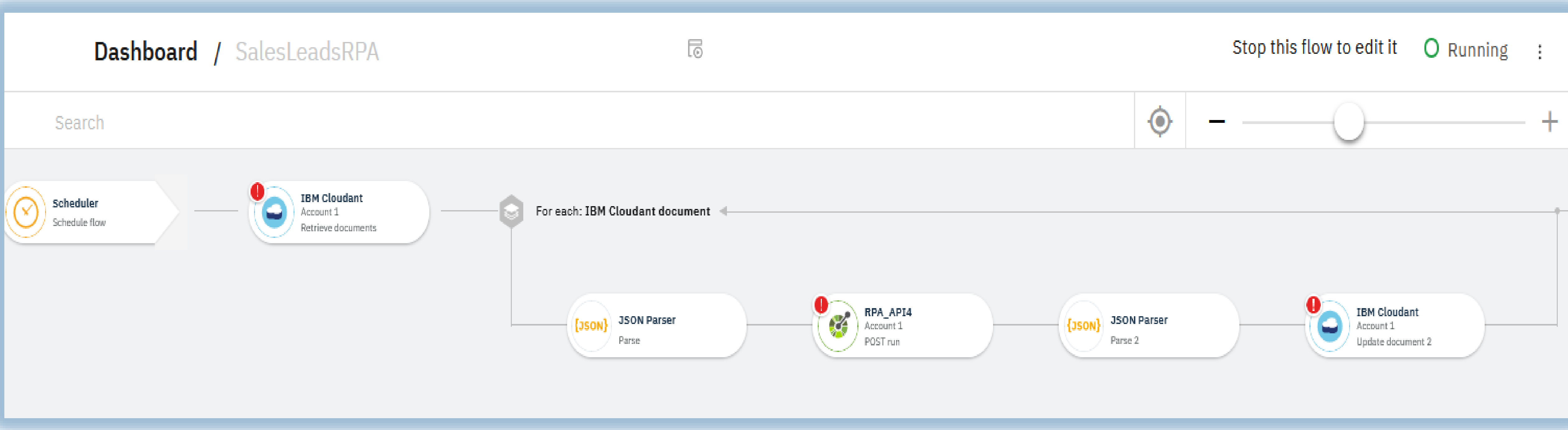
Parameters

Name	Description
mode * required `string` *(query)*	Must be one of INVOKE or RETRIEVE mode
script * required `string` *(query)*	Bot script name to run. This script must be published on the rpaAgentUrl script
rpaAgentUrl * required `string` *(query)*	URL of the RPA agent. If set to LOOPBACK then 202 always returned to simulate a succesful bot run. If a URL, it must point to an RPA agent. It can point to either port 8099 (Direct) or 8096 (Broker). Script parameters are passed in the request body. See documentation https://www.ibm.com/docs/en/rpa/20.12?topic=bot-starting-bots-by-api-ca for more details rpaAgentUrl
unlockMachine * required `string` *(query)*	True if unlock False otherwise unlockMachine
RequestBody * required `object` *(body)*	Default **Example Value** | Model

```
{
        inputParameterName: inputParameterValue
}
```

Parameter content type

Implementation of flow in App Connect



Dashboard / SalesLeadsRPA

Stop this flow to edit it ○ Running

Search

Summary

Advantages

- Build flows to connect to API-less systems
- Use RPA for what it does best – automate human actions and not complex orchestration

Disadvantages

- RPA can already do what App Connect does (but its complex)
- Management of the bot is delegated to App Connect
- Two products means two systems to manage.

End of Presentation

App Connect with RPA Lab