# How API integration Supports Process Automation

Authors: Mark Geene, UiPath General Manager
Charlie Greenberg, UiPath Product Marketing Manager

You've likely heard about the *API economy*. APIs are pervasive, touching nearly every modern application, service, and user.

APIs do more than integrate applications from point A to point B. They also provide direct connections within toolsets, IT, and business webpages. They support developers by making data accessible for building new services—including other APIs.

Think of APIs as the glue that brings together disparate products. They're the foundation for what Gartner calls the "composable enterprise."[1] APIs also play an innovative role in the creation of new automations, including robotic process automation (RPA).

# Expanding the end-to-end automation platform

**According to Gartner, RPA platforms need to address a wider range of automation use cases, including "core integration capabilities to deliver automation via APIs or standard integration methods."[2]**

API connectivity is helping to drive automations powered by artificial intelligence (AI), such as workflow design, user interface (UI) emulation, and document understanding. But to truly automate end-to-end processes, an enterprise automation platform needs to seamlessly combine these services in the following ways:

**Workflow design** orchestrates the steps to complete a task or process. The designer creates automation blueprints based on adapting proven—and often manual—step-by-step business processes or creating new processes from scratch. This approach extends beyond orchestrating APIs, and includes UI automation, document understanding, and even 'human-in-the-loop' steps. Business process examples include:

- Order-to-cash
- Procure-to-pay
- Sales-to-order
- Contact-to-lead management
- Proposal-to-contract processing

**UI emulation** is another key component for integrating applications and automating processes. It allows robots to read your computer screen, select commands, and type within your browser or application. UI emulation can work with any application whether it has APIs or not, while providing access to a wider range of functionality than APIs.

Convenient and less error-prone, UI emulation automates tedious and repetitive tasks directly through the user interface. Graphical user interface (GUI) automation, the essential support tool for screen emulation, can access data when legacy applications fail to provide a comprehensive API.

**Document understanding** helps ensure that software robots can extract, interpret, and process data from PDFs, images, handwriting, and scans.  Powered by artificial intelligence (AI), this capability applies to fixed forms and unstructured layouts and formats. With document understanding, digital documents (such as invoices and purchase orders) are easy to interpret, even when documents are rotated, skewed, or low resolution.

In addition to the capabilities already discussed, leading automation platforms now include:

- API integration capabilities, including prebuilt connectors, facilitating application-to-application workflows
- Document-to-application workflows
- UI-to-API workflows (described later)

# Growing convergence between iPaaS and API automation

**“Focus on a holistic integration and automation platform. Integration and automation are inextricably connected.”[3]**

Analysts often cite the increasing convergence—or coalescence—between integration platform as a service (iPaaS) and API automation. According to Gartner, “. . . by 2024, 90% of integration-platform-as-a-service (iPaaS) vendors will enable process automation, while almost all RPA vendors will offer integration via APIs.”[4] But how much and what kind of integration does an automation platform require? Also, what technology does iPaaS currently offer to support actual process automation?

## Four primary iPaaS integration capabilities

Leading iPaaS vendors continue to build and expand on the following four areas of integration:

### Application integration:

App-to-app integration (or API-based point-to-point integration), can reduce siloed data and is key to synchronizing relevant business information across disparate applications. By offering SaaS-based and on-premises connectors, application integration addresses both future and legacy use cases.

Application integration is the primary use case for iPaaS, with vendors providing low-code tools that support a range of integrator personas, from citizen developers to IT experts.

### Data integration:

Data integration is tied to data quality and data reconciliation initiatives for accurate data warehouse and business intelligence (BI) outcomes. Data integration enables users to request data types or domains from multiple internal and/or external business sources. Once the data is extracted from these sources, it can then be cleansed, transformed, and consolidated into a single, cohesive data set.

### Business-to-business (B2B) integration / Electronic Data Interchange (EDI):

Considered a business partner management solution, B2B integration automates routine (and sometimes highly industry-specific) transactions involving business documents such as purchase orders (POs), invoices, advanced shipping notices (ASNs), and thousands of other document types.

EDI leverages both pre-packaged technology and industry-standard protocols to securely exchange business documents. Typical EDI standards include EANCOM, HIPAA, ODETTE, RosettaNet, SWIFTTM, UNEDIFACT and X12.

It’s worth noting that APIs have become an increasingly flexible alternative to B2B and EDI integration for exchanging business documents.

### API management:

API management has emerged as an important player in modern iPaaS. API management provides an end-to-end platform supporting an entire API lifecycle. From developer portals to validation, testing, secure run-times, and monitoring effectiveness, API management is now a subject of interest for many industry analysts.

In the iPaaS world, API management supports integration artifacts, including data transformations and workflows as APIs—making them consumable by other platforms in an enterprise.

Event-driven architecture (EDA) is a service model that provides strong cohesion between different types of integration, such as application and data integration and siloed segments of microservices.

EDA enables timely sharing of new information and actions based on predefined events. EDA flexibly publishes, captures, processes, and stores events based on actions and data changes across systems. Examples include partner alerts regarding product description updates, changes in marketing promotions, and end-of-business account reconciliation.

## iPaaS and process automation—complementary or competitive?

iPaaS and RPA can coexist within the same organization. For example, UiPath partners with iPaaS vendors who use API integration to start automated workflows in UiPath.

Depending on the complexity of certain automation use cases, companies may want to choose an integration alternative. This is especially true when an organization's automation layer needs to work with both iPaaS, or its older, on-premises integration counterpart, the enterprise service bus (ESB).

Gartner has predicted the near total convergence of iPaaS and process automation by 2024.[4] The two platforms share many capabilities. For example, both iPaaS and process automation offer application-to-application integration capabilities. If two API-connected applications narrowly define a transaction process, iPaaS (and of course, RPA), can easily support that workflow.

Event-driven architectures have a similar use case to API triggers, which generate next steps in an RPA workflow. But why would you want to add even more integration capabilities into RPA? For many organizations, it's an issue of scale. Large enterprises use an average of 175 applications.[5] Each app creates additional steps in a business process while adding more data silos to an enterprise application ecosystem.

**by 2024**

**Gartner's prediction of the near total convergence of iPaaS and process automation**

**175**

**Average number of applications used by large enterprises**

# Using API-patterns in a single automated workflow

Process automation requires a set of API patterns that go beyond traditional API integration. When end-to-end process automation is the goal, it's essential to combine these patterns into a single API-connected workflow:

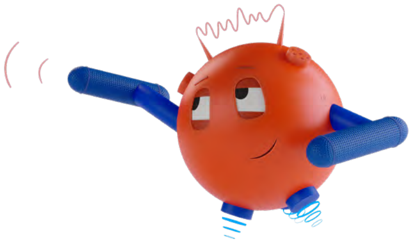
| API-pattern | Description | Key success criteria |
|---|---|---|
| **Application to application** | Process-driven sequence spanning third-party apps, e.g., social media platform lead-to-marketing automation system to CRM opportunity | Event-driven triggers |
| **UI robot to application** | Starts rule-based processes within each step of an automated workflow | AI/ML enabled |
| **Document to application** | Robots read/decipher all application documents, such as invoices or purchase orders-- and take action | Using AI/ML, robots understand any document form, including handwriting |
| **Human in the loop** | Robots assign process tasks to human team members for approvals, escalations, and exceptions | Central monitoring console, ability to combine multiple processes |
| **Attended/unattended** | Unattended robots support an entire workflow without human intervention, and attended robots work in tandem with humans—essentially sharing tasks | Unrestricted access to all third-party systems<br>Hybrid attended/unattended capabilities, multiple deployment models |

# Unifying UI and API integration

Based on a survey of 118 business technology decision makers, Forrester Research determined that 95% of organizations rank process optimization on a scale from important to critical.[6] Automation is a top concern because of its role in areas such as digital transformation, improved customer experiences, cost reduction, and higher accuracy. But when the goal is to replace paper-based, manual tasks and workflows, enterprise automation requires more than just APIs.

By combining and embedding API integration with UI automation, RPA  offers the richest capability set for enterprise automation. In fact, API integration is now essential to designing, building, and running complex and scalable automations— including the ability to seamlessly design and combine UI automations, API integrations, and AI-based document understanding in a single workflow.

# Four essential API integration features—and best practices for end-to-end automation



### 1. Ease of authentication:

If RPA developers can't connect, they can't automate. Connectivity should be easy to perform and authorize from any point within the automation platform.

- Design vs. run-time connections that support sandbox connectivity should be shareable within an organization, while allowing individual users a personal connection to run automation.
- Support for both personal and service accounts is required.

### 2. Robust connector resources:

A catalog of prebuilt and customizable connectors will significantly simplify API integration by containing the necessary native objects and standardized activities associated with a given application. Components include:

- No-code connector development to quickly fill gaps in a connector catalog
- Centralized governance that enables developers to centrally manage and validate connections, as well as enable and disable access
- Re-usable templates to accelerate productivity
- The ability to invoke any method through APIs and applications, including the REST-HTTP method, as in:

  GET = read data, POST = insert data, PUT or PATCH = update data, or insert, for example, when a new id is updated for a new account object in SFDC (salesforce.com)

**Note:** Many connectors provide dynamic application data models, supporting the creation of custom fields and custom objects. Robust connectors go even further by enabling discovery of custom objects and customizable data fields, or properties, at design-time.

### 3. Event triggers:

Server-side triggers automatically empower robots to work across dozens of applications based on events in connected systems. Server-side events include data updates, insertions or added data, and deletions.

For example, adding a new employee to an HR system could trigger a series of tasks for IT, security, compliance, and payroll. Events can be monitored with webhooks that notify the workflow to execute or ask for record changes by a polling framework. Both are important to automation programs, as most API endpoints still can't identify changes within an application.

### 4. Single API/UI-based design environment:

To maximize productivity, developers should have access to API and UI-based capabilities (including API integration) within the same integration design environment, or both integrations should be available within the same automation or workflow.

The platform should provide separate design environments, supporting different, and clearly defined developer skillsets, including:

1. RPA & Integration developers/IT specialists capable of building advanced, cross-platform automations
   - Specifically, the skills to code C# projects, as well as test automations for web, desktop, and mobile apps
2. Citizen developers and/or business users whose productivity relies on the ability to configure simple robots and native integrations
   - Enabled through no-code tooling, drag-and-drop editors, and prebuilt wizard templates
3. An integration team working with a Center-of-Excellence (CoE)
   - Including business and IT application owners

# Bringing it all together – UiPath enterprise automation examples

The following real-world automation examples are built or conceived through the UiPath end-to-end automation platform. Each workflow process design incorporates:

- Prebuilt third-party application connectors with connection authorization and selected from the UiPath Integration Service Connector catalog
- Event-driven server-side triggers that activate robots to take the next step in each process (for example, initiating a workflow following an update in Salesforce, a posted transaction in SAP, or the receipt of an email)
- Combined UI and API automation available in a single UiPath Studio design environment, supporting all workflows

## Applicant-to-hire

| Step 1 | Step 2 | Step 3 | Step 4 |
|---|---|---|---|
| Robot automatically creates a new user in Salesforce. | Trigger event kicks-starts a service request in ServiceNow. | Robot logs into a proprietary application that doesn't have an API by using UI automation. | Robot drafts a summary email to the new hire via Microsoft Outlook, using an API-native integration. |

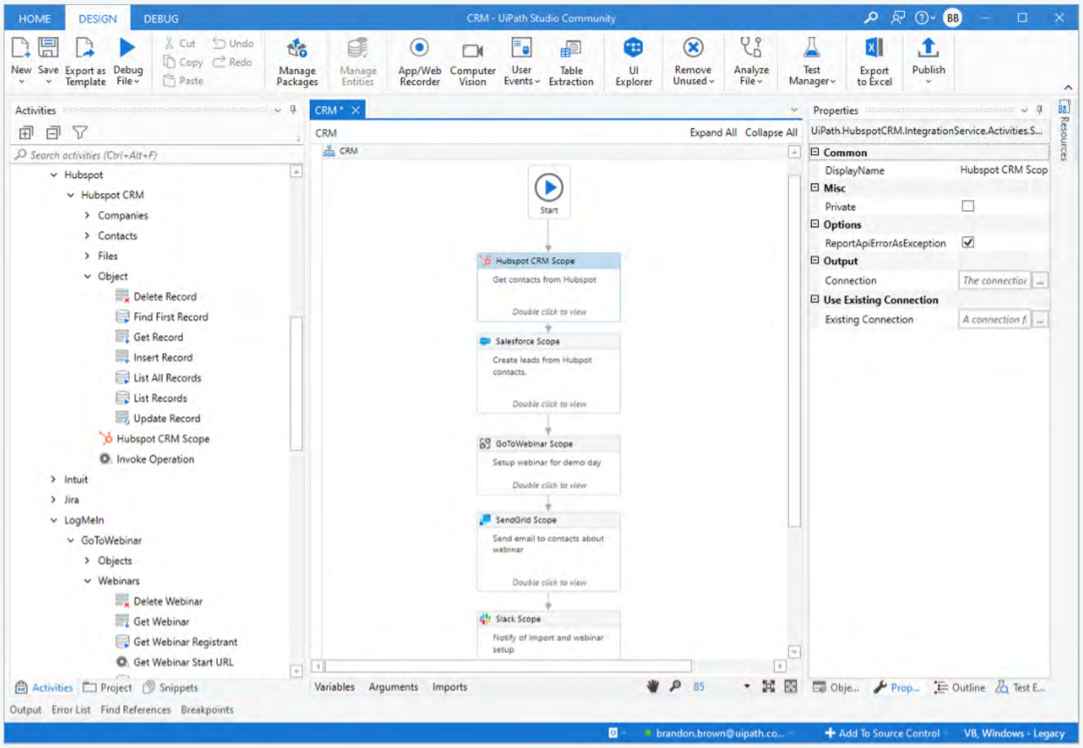
## Contact-to-lead

| Step 1 | Step 2 | Step 3 | Step 4 |
|---|---|---|---|
| A robot automatically creates a new customer contact in Hubspot CRM. | Oracle Eloqua localizes and personalizes marketing content for the new customer contact. | Oracle Eloqua moves the customer contact into marketing automation. | The Robot starts an email (part of a nurturing campaign) for the new customer contact in Marketo. |

## Contact-to-webinar attendee

HubSpot | salesforce | GoToWebinar | SendGrid | slack

| Step 1 | Step 2 | Step 3 | Step 4 | Step 5 |
|---|---|---|---|---|
| HubSpot CRM provides webinar contacts to Salesforce. | Salesforce creates leads from identified CRM webinar contacts. | GoTo Webinar stages the marketing webinar for attendees. | SendGrid emails invitations to contacts, promoting webinar attendance. | Slack enables near real-time webinar communications. |



CRM

Start

**Hubspot CRM Scope**
Get contacts from Hubspot
*Double click to view*

**Salesforce Scope**
Create leads from Hubpot contacts.
*Double click to view*

**GoToWebinar Scope**
Setup webinar for demo day
*Double click to view*

**SendGrid Scope**
Send email to contacts about webinar
*Double click to view*

**Slack Scope**
Notify of import and webinar setup

Properties
UiPath.HubspotCRM.IntegrationService.Activities.S...

| | |
|---|---|
| **Common** | |
| DisplayName | Hubspot CRM Scop |
| **Misc** | |
| Private | □ |
| **Options** | |
| ReportApiErrorAsException | ☑ |
| **Output** | |
| Connection | *The connectior* |
| **Use Existing Connection** | |
| Existing Connection | *A connection f* |

brandon.brown@uipath.co...

UiPath Studio environment depicts an end-to-end marketing automation process where each step is event-driven by robots across five different, third-party systems.

# Takeaways

We now count on automation to seamlessly blend together any tool you need to automate all aspects of a process.

To achieve this, APIs and UIs are working together as equal partners.

Combining both UI and API automation streamlines integration while expanding the scope for new automations. The ability to leverage a robust catalog of prebuilt connectors that support vital customer applications encourages ongoing automation innovation. In UiPath Integration Service, UI and API automation capabilities converge into a single design and runtime tooling.

**Watch this demo to see how easily you can combine both UI and API automation—within the same development environment—to create an order management process connecting Salesforce and ServiceNow.**

1. *Gartner: "Future of Applications: Delivering the Composable Enterprise. Deliver innovation and adapt more quickly", Analyst(s): Dennis Gaughan, Yefim Natis, Gene Alvarez, Mark O'Neill; Published: 11 February 2020

2. *Gartner, "Beyond RPA: Build Your Hyperautomation Technology Portfolio", Saikat Ray, Paul Vincent, et. al., 22 October 2021

3. IDC Worldwide Integration and API Management Software Forecast, 2021-2025, Shari Lava.

4. *Gartner, Market Share Analysis: Robotic Process Automation, Worldwide, 2020, Fabrizio Biscotti, Cathy Tornbohm, et al., 26 May 2021

5. Okta Businesses at Work 2021; WSJ, Employees are Accessing More and More Business Apps, Study Finds, 2019

6. Forrester's Q2 2021 Global Digital Process Automation Vision Survey

* Gartner, Market Share Analysis: Robotic Process Automation, Worldwide, 2020, Fabrizio Biscotti, Cathy Tornbohm, et al., 26 May 2021 Gartner does not endorse any vendor, product or service depicted in its research publications, and does not advise technology users to select only those vendors with the highest ratings or other designation. Gartner research publications consist of the opinions of Gartner's research organization and should not be construed as statements of fact. Gartner disclaims all warranties, express or implied, with respect to this research, including any warranties of merchantability or fitness for a particular purpose.