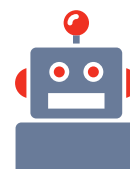


マッシュアップで業務を有益化

業務アプリ視点での RPA 開発

RPA ツールが中心の開発

独自の開発手法、独自のプラットフォーム
独自の AI 連携、独自の運用
他のサービスは RPA 側から API 連携



業務アプリ中心の開発

既存の財産が活かせる
RPA のロボットを業務アプリから動かせる
API で多種多様なサービスを連携できる (AI、IoT 含む)



現在使用している業務システムは、企業にとって財産です。

RPA が流行る前から、企業は業務処理の自動化に努めてきました。業務アプリケーションの基幹となる部分は、データをインプットすれば、処理されてアウトプットされます。そうした中で、ヒトが行っていた PC 操作などの作業をロボット化したいために RPA の需要が高まったのです。

多くの RPA ツールは、それ自体がプラットフォームのように AI や ERP などと融合して、簡単にロボットが作成できる先進のプログラム言語であるかのように別の世界観で進化してきました。そのため、既存の業務アプリに手を加える場合、新たに RPA ツール側も別作業で、UI の部分を変更しなければなりません。それでは、新たに RPA ツール側の開発方法で、業務アプリを作成し直す必要があります。

一方、既存の業務アプリから、API を利用してロボットを動かすという方法もあります。これが、いわゆる開発型 RPA とされる手法で、汎用的なプログラミング言語を用いて、プログラム上にロボットの動かし方をコーディングして、API を通して RPA のソフトウェアロボットを動かします。この方法だと、業務アプリの方が中心となり、PC 操作などの UI の部分を API でそのプログラムに追加するだけなので、別途ロボットの動きを管理する手間が軽減でき、従来の運用方法でシームレスにソフトウェアロボットを取り込むことができます。

両者、一長一短がありますが、RPA をスクラッチで開発するのではなく、過去の財産や開発ノウハウを有効活用するには、従来使用してきた開発言語でロボット開発の部分も取り込んだ方が、開発も運用も楽な場合が多いです。

近年、コンテナなどを利用したマイクロサービスが盛んになっていますが、昔から「マッシュアップ」というネット上の複数のサービスを組み合わせて、あたかも一つのサービスとして提供する手法があります。汎用的言語で開発している方が、クラウドの MAP サービスなど、すでに公開されている API も効果的に使用することができます。たとえば、ProgrammableWeb では、数万という API についての情報や、連携方法のチュートリアルなどが公開されています。こうしたものを参考に、ビジネスにとって有効なサービスを API で取り入れ、AI や IoT と同様、RPA のロボットについても、ひとつの組み入れるべき部品としてプログラミングにより API 連携すれば、圧倒的に短期間に有益な自動化された業務アプリを開発することができます。