

産業用アプリケーションにおける

Python

Source: [PLCnext Community Business Lounge \(Python in Industrial Automation \(plcnext-community.net\)\)](https://plcnext-community.net)

Yuri Chamarelli, Lead Product Marketing Specialist

産業用オートメーションにどのプログラミング言語を使用するか考える時、通常は、古典的なラダーダイアグラム (LD) や構造化テキスト (ST) などの PLC 用の IEC 61131-3 言語にある言語から始めます。ロボット工学のようなアプリケーションでは、C 言語ファミリーのようなコンパイル方式の低レベルプログラミング言語をよく目にします。

ごく最近まで、工場の現場で Python を使おうなんて話は誰もしていませんでした。何か変わって、インタプリタ方式の高レベル言語が急に魅力的になったのでしょうか？ Python のコードは一般的にマシンコードに近いところで動作するコンパイル済みの言語よりも動作が遅く、産業用システムを開発する際の重要な点であるリアルタイム制御のタスクには対応していません。

しかし、私たちは今、Python にスポットライトを当てている要因の光がどんどん合わさって強くなってきているのを目の当たりにしています。第一に、インダストリー4.0 は、産業オートメーションについて考え方のパラダイムを変えてようとしています。自律性の向上、豊富なビッグデータの活用、アディティブ・マニュファクチャリングやクラウド・コンピューティングのような次世代技術との完全な統合を伴う「スマート」な機械に重点を置いています。

インダストリー4.0 のもう一つの意味は、産業用モノのインターネット (IIoT) です。ローカルネットワーク上で産業用機械を接続し、リアルタイムに M2M (マシン・ツー・マシン) 通信を行い、分析のためのセンサーデータを安定的に提供するものです。その結果、私たちは IT と OT の融合を目の当たりにしており、これまで IT (情報技術) の専門家と OT (操作技術) の専門家を長い間隔でいた壁が崩壊しつつあります。そして、IIoT デバイスは情報を利用して機械操作を最適化します。

ここで Python の話に戻ります。世界で最もポピュラーなプログラミング言語である Python の強みを見てみると、IIoT にとっての明確なメリットがいくつか見えてきます。第一に、Python は巨大なデータセットを扱うことができることで有名です。第二に、Python の発明者である Guido Van Rossum は、高い可読性を目的として Python を設計しました。これは、複数のエンジニアが同じコードに取り組んだり、保守したりする際に重要な特徴であり、イノベーションがイノベーションを呼ぶサイクルを生み出す要因にもなります。そして最後に、Python はオープンソースであり、素晴らしいコミュニティを持ち、今日の最も魅力的なアプリケーションの多くに利用されています。

さらに見ていきましょう。

■機械学習(ML)

間違いなく、Python が最も影響力を発揮している分野は、機械学習(ML)です。明示的なルールを誰もコーディングすることなく、データからアルゴリズムが自動学習する AI の一分野です。よく利用されている産業アプリケーションとしては、予知保全や自律型ロボット工学などがあります。今日の ML のほとんどは Python で書かれています。PyTorch や Google のオープンソース TensorFlow のようなフレームワークは Python を使用しています。Amazon のクラウド AI サービスである AWS SageMaker には、Python ソフトウェア開発キット (SDK) が組み込まれています。簡単に言うと、Python は ML では最高のツールです。

一般的な産業用 ML のセットアップは以下のようになります。機械とセンサーのデータはクラウドに送られますが、そこでは ML モデルを訓練するために使用できる高性能なリソースに簡単にアクセスできます。例えば、データを分析して機械がどれくらいで壊れるかを知ることができるモデルのように、モデルを訓練できたら、そのモデルを工場に送り返すことができます。



生産デバイス自体に GPU を埋め込んでエッジコンピューティングを実行するか、ローカルにアクセスできる IIoT ゲートウェイのリソースをフォグコンピューティングに利用することで、訓練されたモデルを現場で利用することができます。自律型ロボットを見てみると、Python コードは高レベルの目標を解釈し、それがハードウェアと直接インターフェースする低レベルのコンパイルされたコードによって動作として解釈されます。人が腕を動かす時もこれに似ています：脳が高レベルの意図を設定し、低レベルの神経系が筋肉を動かします。インダストリー 4.0 の世界になっていき、製造業者が AI のもっと革新的な用途を見出すようになると、Python のエンジニアリングスキルはますます重要な資産になるでしょう

■コンピュータビジョン(CV)

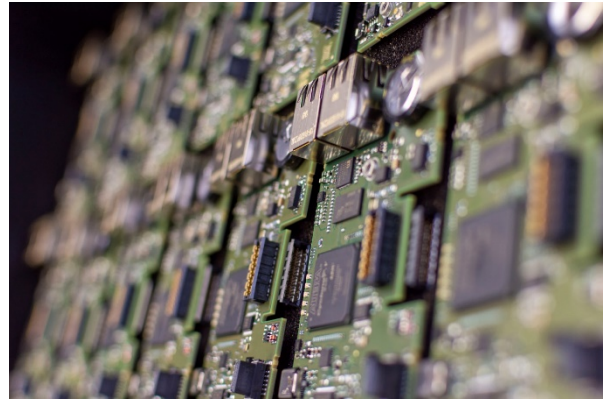
ロボットアームが何かを拾うためには、まず何を見ているのかを知る必要があります。コンピュータビジョン (CV) は、機械がカメラを目として使うことを可能にしますが、もっと重要なのは、見ているものの認識を可能にする AI 分野であることです。ML における Python の大きな役割を考えれば、Python が CV にどのように役立つのかは想像に難くありません。

元々 90 年代後半に Intel によって開発された OpenCV は、現在ではオープンソースの CV 開発に最適な選択肢の一つとなっています。ライブラリはまだ C++ で書かれていますが、Python ラッパーである OpenCv-Python は、元の C++ コードの速度を維持しつつ、Python の利点を引き出しているため、CV においてディープラーニングのような ML アプリケーションに理想的です。

さらに、OpenCv-Python は NumPy の配列を出力として生成するので、データをすぐに SciPy や Matplotlib などの Python ツールや他の ML プラットフォームに渡すことができます。その結果、機械は、ピクセル以上のものを見ることが可能になり、製品を区別したり、品質保証のチェックをしたり、環境を複雑かつ微細に操作したりすることができるようになります。

■エコシステムをつなげる橋をつくる

機械メーカーが製品を出荷する際、通常、ヒューマン・マシン・インターフェース（HMI）以外との通信機能は優先されません。多くの機械が独自のコードやGコードをハードウェアに非常に近いところで実行しているという事実を加味すると、IT/OT融合のためのIT側処理はさらに難しくなります。機械が同じ言語を話すことができない場合、どのようにして機械同士を会話させられるでしょうか？



私たちには翻訳者が必要であり、Pythonがその仕事を引き受けてくれます。OpenMTCのようなプログラムは、M2MやIoTアプリケーションのためのミドルウェア、「ソフトウェアの接着剤」として機能します。例えば、このミドルウェアをRaspberry Piのようなシンプルなデバイスに入れると、Pythonスクリプトはあるソースからデータを受け取ると、別のマシンにそれが読めるフォーマットに変換して送信します。

簡単な例としては、温度に敏感な製造プロセスがあります。機械自体は温度を調整できないかもしれませんが、温度を読み取り、特定のしきい値を超えた場合、ミドルウェアに知らせることができるかもしれません。するとミドルウェアは、ヒーターにスイッチを切り替えるように指示することができます。

同じ理屈は、他の機械の出力に依存しているどの機械にも適用できます。さらに、現在の機械を統合して性能を向上させるだけでなく、新たな可能性の世界への扉を開くことができます。その一つが、ドライバーレス自動車です。他の車と通信することで、交通パターンを最適化し、通勤時間を短縮し、事故を減らすことができます。Pythonは、そのように相互接続された未来のキービルディングブロックになると言えます。

■まとめ

最終的にはPythonを使ってロボットハードウェアを制御したり、製造機械との直接のインターフェースを行ったりするようにはならないかもしれませんが、だからといってこのプログラミング言語が産業用のアプリケーションで使えないということにはなりません。Industry 4.0の全体像を見てみると、データがその特徴であることがわかります。そこではデジタルの世界を物理的な世界に直接重ね合わせています。だからこそPythonが必要なのです。2つの世界間のギャップを埋めるために、前例のない大量のデータを処理するために、そしてマクロ的な制御をサポートするためにPythonが必要です。

本資料および PLCnext Technology の詳細は URL: <http://www.plcnext.jp> まで

本資料に掲載している文章、写真等の著作権は当社に帰属し、事前の許可無く無断で複製、転載、転用することは著作権法で禁止されています。発行元：フエニックス・コンタクト株式会社