

学務システムにおける自動操作プログラムの適用

Apply to Automation Procedure Program for Educational Affairs System of University

成瀬 健一郎

山田 洋巳

Kenichiro Naruse

Hiromi Yamada

〈摘 要〉

工場などの作業員に対応する労働の代替えとしては、ロボットなどを使って自動化がされてきた。現在、政府主導の働き方改革に対応して、事務作業に対する自動化として、Robotic Process Automation (RPA) ツールが注目されるようになった。これは、ソフトウェアの入力やマウス操作などを決められた手順や条件に応じて自動で行うものである。本論文では、RPA ツールと自作の自動操作プログラムを使って学務システムの自動操作を行い、それぞれを利用して自動操作を行った場合の評価や問題点についてまとめたものである。

〈キーワード〉働き方改革 RPA 学務システム 省力化 自動操作プログラム作成

I. はじめに

政府主導の働き方改革において、時間外労働の上限規制、年次有給休暇の確実な取得などが示された^[1]。時間外労働の上限規制の対策として、労働人口減少に伴う人手不足の中、人を雇い入れて一人当たりの時間外労働の時間を減らす手段は、企業として採ることは難しい。そこで、人を雇い入れずに、業務改善を行い同じ仕事を短時間で行う仕組みづくりの1つとして、コンピュータの自動操作が考えられる。これは、コンピュータで行われるルーチンワークについて、キーボードやマウス操作を、人間に成り代わってコンピュータに操作を行わせようとするものである。コンピュータ自身にコンピュータ上のソフトの操作を行わせることによって、ルーチンワークにおける正確さとスピードの両方を追求し、人間をもっと創造的な仕事に集中させるとともに、働き方改革で示された時間外労働時間の遵守や、労働者人口の減少に伴う労働力不足などの解決にも貢献できると考えられる。コンピュータ上のソフトの自動操作を行うためには、Robotic Process Automation (RPA) ツールを利用するか、自動操作プログラムを作成するかの2通りの方法が考えられる。本論文では、自動操作プログラムについての概要、RPA ツールの種類、特徴をまとめ、さらに本学の学務システムから成績証明書を発行する自動操作プログラムを、

RPA ツールと、PowerShell を使った自作の自動操作プログラムの両方を比較し、RPA ツールを使って自動操作を行った場合と、自動操作プログラムを作成して自動操作を行った場合について評価を行い、それぞれの問題点について論ずる。

II. 自動操作プログラムの概要について

コンピュータシステムにおいて、処理する内容が1つのシステム内で収束するように作られ、かつ、データを適切に受け渡しされるような仕様にてプログラムが作られていれば、RPA ツールや自動操作プログラムは必要ない。部分的に市販のソフトウェアを使ったり、連携する予定ではなかったシステムやプログラムと連携したりすることになった場合、データのやり取りを行うために、人間による処理の介入や自動操作プログラムが必要になってくる。例えば、文字認識ソフトで文字認識を行い、それをデータベースに登録する処理について考える。文字認識ソフトは、かなり複雑な処理が必要になってくるため、開発を依頼した場合は非常に高額なソフトになってしまうであろう。そこで、パッケージソフトとして販売されている一般的な文字認識ソフトを導入すれば、安価にその機能を手に入れることが出来る。しかし、一体として開発されたものではないので、パッケージソフトの文字認識ソフトで文字認識をした後、データベースにそのデータを人の手によって転送するという処理が必要になってくる。こういったところで、自動処理プログラムを動作するようにしておけば、文字認識ソフトとデータベースの間で人間が介入しなければ継続できないようなエラーが発生しない限り、半永久的に動き続けることができるため、確実に省力化を図ることができる。

III. RPA ツールについて

RPA ツールとは、人間に成り代わってコンピュータ上のソフトウェアを操作するものである。事前に、このウィンドウ名のウィンドウの、この位置をクリックする、条件に応じて異なる処理を行う、処理を繰り返すといったことを設定しておけば、設定通り操作される。また、いくつかの RPA ツールは、カメラで録画するように動作指示を記録することができるため、全く同じ処理であれば、簡単に RPA を使用することが可能である。ただし、ほとんどの RPA ツールは RPA ツールが動作中のコンピュータに対して、人間が操作を行うと RPA ツールが誤動作を起こしてしまうため、RPA ツールの動作中には人間は操作ができない。

IV. 自動操作プログラムについて

自動操作プログラムについては、ComObject 対応のソフトウェアであれば、word や excel などのマイクロソフトオフィスに含まれるソフトのマクロ機能、PowerShell、VBScript や C 言語などを使っても、開発することが出来る。なお、ComObject 対応のソ

フトはマイクロソフトオフィス以外にはほとんどない。従って、マイクロソフト社製品であれば、マクロを使って ComObject 対応のソフトはある程度の自動操作を行うことができる。しかし、これを実現させるためには、相当なプログラミングの技術と、作成する時間が必要となり、現在一般的に使われている RPA ツールを使用して動作させる方法とは大きく異なる。

なお、マイクロソフトオフィス等に搭載されているマクロとの違いは、マクロは自らのソフトの動作については処理ができるが、ComObject 対応のソフトを除いて他のソフトの操作は行えないことである。

V. 主な RPA ツール

RPA ツールは、現在、個人使用や小規模企業においては無料のものから数千万円といった高額のものまで存在している。代表的なものを表 1 にまとめる。

VI. 自動操作プログラムの優位性

1 RPA ツール

ほとんどの RPA ツールで、自動操作させたい画面操作をビデオカメラのように記録し、再生することができる。記録・再生方式が採れるため、高度なプログラミング知識がないユーザーでも、比較的容易に自動操作プログラムを作成することができる。また、編集もできるため、一度操作を記録し、それを編集して、繰り返し実行するように変更するといった指示も容易にできる。なお、自動操作の流れをグラフィカルに表示するものもあり、画面操作した手順を目で見て確認することができる。

表 1 主な RPA ツール

名 称	UiPath ^[2]	WinActor ^[3]	UWSC ^[4]
無償または有償	非営利団体、研究・教育機関、小規模事業者に向けて一部のツールを無料で使用できる。	無償での配布はない。	基本機能は無料で使用可。単体で動作する実行プログラムの作成などの機能を使いたい場合は、有償。
使い方の説明	ビデオチュートリアルや日本語ユーザーガイド、オンライントレーニングが提供されている。	無償では配布されないため不明。	オンラインマニュアルのみ。
公共性の高い使用ユーザー	早稲田大学、多摩市、別府市、なんと南砺市、加賀市、茨城県、伊予銀行、三井住友信託銀行株式会社等	埼玉・幸手桜高校、立命館大学、みずほフィナンシャルサービス、NTT DATA 等	不明
課金方法	有償のものについてはサブスクリプション型（毎月支払う）	サブスクリプション型（毎月支払う）	有償のものについては売り切り。但し、現在作者と連絡がつかない状態になっており購入不可。

2 自動操作プログラム

今回のような自動操作プログラムを作成するためには、有償の開発環境を使うこともできるが、ほとんどが無償の開発環境で作成できると考えられる。PowerShell であれば、Windows10 に標準の開発環境で開発できる。また、ウィンドウがどこに表示されても、画面内の ID やオブジェクト名を使って処理を指示するため、誤動作する可能性が低い。また、RPA ツールではできないような複雑な処理もプログラム次第で作成することができる。

VII. 自動操作プログラムでの問題点

1 RPA ツール

自動操作のプログラムを作成する時間や自動操作プログラムが起動するまでの時間を考慮すると、行うべき一定量の処理がなければ RPA ツールを使うメリットは低い。また、RPA ツールが有償の場合、コストパフォーマンスの面において、その操作を RPA ツールに任せただけの場合、どれだけのコスト削減が可能になってくるか評価する必要がある。RPA ツールは買い切りではなく、年度ごとの使用ライセンスが必要であることが多いので、年度内にライセンス費用以上のコスト削減効果を出す必要がある。また、セキュリティの面から考えると、RPA ツールでは別画面からの転記を繰り返すような操作を容易に作成できることから、一画面ずつしか見れなくなっている情報であっても、RPA ツールの自動操作によってすべてのページを参照し、参照したページのデータをローカルのデータベースに転記して持ち出すなど、情報漏洩につながる危険性も考えられる。

2 自動操作プログラム

画面操作を記録して再生するような機能はないので、1 アイテム、1 画面に対応したプログラムを書く必要がある。これには、相当なプログラミング技術と時間が必要となる。また、自動操作プログラムは、被操作対象のプログラムと密接に関係して動作するため、被操作プログラムがバージョンアップなどで変更された場合、RPA ツールよりも細かにプログラムの修正を行わなければならない。

VIII. 実際の利用について

学務システムは、WEB アプリケーションとして動作するシステムになっている。学務システムから成績証明書を 1 枚だけ発行する操作の流れは、Appendix 1 の通りである。成績証明書を印刷する場合、パラメータとして、学籍番号、発行番号の有無、履修中科目の印刷の有無、不合格科目の印刷の有無、発行日など、たくさんのパラメータを指定する必要がある。これを毎回、間違えずに指定を行う必要性があり、証明書発行に時間がかかっている原因の一つになっている。これを解決するために、Appendix 1 の通り自動操作を

行う RPA ツールの設定^[5]及び、PowerShell を使った自動操作プログラムの 2 種類の自動化プログラムを作成・実行を行い、自動操作を行う際の評価及び問題点について 6.1 及び 6.2 において示す。

1 RPA ツール

(1) 評価

無償利用が可能な RPA ツールを利用して評価に使用する。自動化の手順は、まず、成績証明書が発行されるまでの画面操作を記録した後、ログイン処理、対象となる学籍番号を入力ボックスに変更、動的に変更されるウィンドウ名をワイルドカードに変更を行った。RPA ツールを使った成績証明書印刷の自動操作の開発を行った端末では、学籍番号をウィンドウに入力するだけで印刷が可能となる。RPA ツールでの自動操作の設定も非常に簡単に行うことができる。

(2) 問題点

開発を行っていない端末において成績証明書印刷を RPA ツールによって操作をさせるためには、無償使用の制限上、実行プログラムが作成できない。従って、RPA ツールを動作させるためには、成績証明書の発行指示を行う端末全てに RPA ツールをインストールする必要がある。RPA ツールは繰り返し処理を複数回行うことを省力化することを目的として作られており、起動・終了を繰り返すことを想定されていないためか、RPA ツールの起動に約 40 秒かかるケースもある。従って、今回行った成績証明書を 1 枚だけ印刷する処理には向かない。また、費用を払えば、実行形式も作ることが出来るが、費用対効果が悪い。

2 自動操作プログラム

(1) 評価

成績証明書を 1 枚印刷する場合、RPA ツールの優位な点である複数のアプリケーションを切り替えての操作が必要ではない。また、Web 操作のみで証明書を発行できることに着目し、PowerShell から ComObject を使って自動操作を行うプログラムを作成する。プログラムは、Windows10 Enterprise 上の Windows PowerShell ISE を使って作成する。

被操作側の web が自ら変更や修正ができるものであれば、操作するエレメント（項目）に ID を入れることが出来るので、比較的簡単に自動操作プログラムが作れる。しかし、今回ターゲットとなる被操作側の web システムは外部で作成されたものであり、プログラムソースも非公開であるため、操作するエレメントに ID を埋め込むなどの対応は不可能である。従って、ウェブブラウザに実際に送られてきた HTML ソースを解析しながらプログラムを作成する。

(2) 問題点

PowerShell からフレームを使って作られた web 画面を操作するためには、操作するフレームを指定する必要がある。このためには、HTML ソースを調べる必要がある為、作成に手間がかかる。さらに今回、操作を行った学務システムの成績証明書発行の画面では、フレームの中にフレームを使うといった構造になっていて、オブジェクトの指定が複雑になる。また、PowerShell ISE において、getElementsByTagName や、getElementByID などの命令が一度目は動作するが、二度目が動作しない場合がある。これは、同等のコマンドの IHTMLDocument3_getElementByID を使って回避する。

IX. 結論


Web アプリケーションである学務システムの自動操作を、RPA ツールと自動操作プログラムを作成するという 2 つの方法で、学務システムの web 画面から成績証明書の印刷を自動操作で行う自動操作プログラムを動作させた。RPA ツールを使った方法では、簡単に自動操作の設定が可能となり、使用することができた。しかし、繰り返しではない処理に RPA ツールを使うということは、プログラム起動のオーバーヘッドが大きいため、起動してしばらく使い続けるものには適しているが、短時間で処理が終了してしまうものには向かない。次に、PowerShell を使った方法では、プログラム作成がかなり複雑で作成に時間がかかった。特にフレームを使った web アプリケーションの場合、操作するエレメントの指定をどのフレーム位置か正確に指定する必要があり、フレームの中にフレームを入れるといった使われ方をしていると、プログラム作成は非常に困難なものになる。しかし、プログラムの起動や動作は素早い為、間欠的に使用する自動操作プログラムの場合は、RPA ツールより自動操作プログラムの方が向いている。

RPA ツール、及び自動操作プログラムを使った自動操作は、人間をコンピュータの単純な操作から解放してくれるが、自動操作させる手順について事前に記録をさせたり、プログラミングをしたりして、自動操作させる仕組みづくりを人間が行う必要がある。また、最低限のプログラミングの知識がないと、自動操作プログラムではもちろんの事、RPA ツールにおいても適切な条件分岐や繰り返しの処理を指示することは難しい。しかし、RPA ツールは自動化処理の作成が非常に簡単で、視覚的に操作の指示が行えるため、単純で細かなエラー処理などを処理しなければ、数分で作ることができる場合もあり、初心者にも使いやすい。


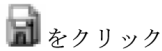
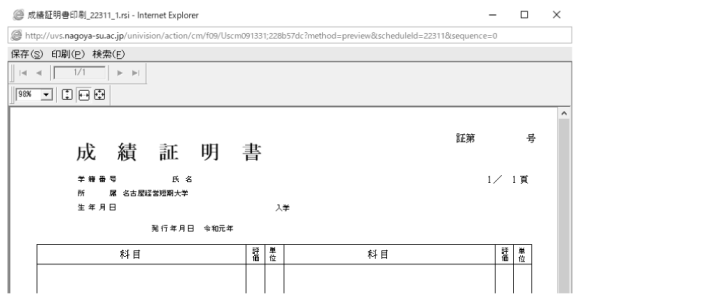
Appendix 2 において、PowerShell を使った web の自動操作についてのサンプルを掲載した。サンプルでは HTML のフレームの中に入れ子となったフレームに対応した操作や、JAVA Script から生成されるダイアログの表示の抑止を行っており、PowerShell で自動操作を行う場合のひな型とできるので、これを参考に、自動操作プログラムを構築していくことが出来るであろう。

Appendix 1 web 操作の流れ

	<p>ユーザー名とパスワードに適切なデータを入れる</p>
	<p>帳票印刷をクリック</p>
	<p>成績証明書印刷をクリック</p>

	<p>発行番号 なし</p> <p>履修中を出力するか しない</p> <p>学籍番号 該当学生の 学籍番号</p> <p>下側にある実行をクリ ック</p>
	<p>OK をクリック</p> <p>(自動処理プログラムの 場合は、このダイヤ ログの出力を抑止す る。)</p>
	<p>実行結果一覧画面へ をクリック</p>

【研究ノート】学務システムにおける自動操作プログラムの適用

	
	<p>印刷をクリック</p>
	<p>OK をクリック</p>

Appendix 2 自動操作プログラム（一部はセキュリティのため、文字列を消してある。）

```

if ($args[0] -eq $null) {"学籍番号が指定されていません";exit}
$stu_id=$args[0]

if ($args[1] -eq $null) {$sw_maisu=1}
$sw_maisu=$args[1]

$shell = New-object -ComObject "Shell.Application"
$ie = $shell.Windows() | where {$_.name -eq "Internet Explorer" -and ($_.LocationName -eq "名古屋産業大学 UniVision" -or $_.LocationName -eq "職員トップページ")}
if ($ie -eq $null) {
    $ie = New-Object -ComObject InternetExplorer.Application
    $ie.top=0
    $ie.left=0
    $ie.height=900
    $ie.width=1020
    $ie.visible = $true
    $url="
    $ie.navigate($url)
    while ($ie.busy -or $ie.readystate -ne 4){ Start-Sleep -Milliseconds 1000 }
    $u_name=[System.Security.Principal.WindowsIdentity]::GetCurrent().name
    $u_name=$u_name.replace("¥","")
    $u_pass =(Read-Host パスワード -AsSecureString)

    $BSTR = [System.Runtime.InteropServices.Marshal]::SecureStringToBSTR($u_pass)
    $PlainString = [System.Runtime.InteropServices.Marshal]::PtrToStringBSTR($BSTR)
    [System.Runtime.InteropServices.Marshal]::ZeroFreeBSTR($BSTR)
    ($ie.document.IHTMLDocument3_getElementByID("j_username")).value()=$u_name
    ($ie.document.IHTMLDocument3_getElementByID("j_password")).value()=$PlainString
    ($ie.document.IHTMLDocument3_getElementsByTagName("input") | where{ $_.name -eq "login"}).click()
    while ($ie.busy -or $ie.readystate -ne 4){ Start-Sleep -Milliseconds 200 }
    sleep -Seconds 1
    do {
        start-sleep -m 200
        $ie = $shell.Windows() | where {$_.name -eq "Internet Explorer" -and $_.LocationName -eq "名古屋産業大学 UniVision"}
    } while($ie -eq $null)
} else {
    $url="
    $ie.navigate($url)
}
while ($ie.busy -or $ie.readystate -ne 4){ Start-Sleep -Milliseconds 200 }
$doc = $ie.document
@($doc.frames.item([ref]1).document.IHTMLDocument3_getElementsByTagName("a") | where {$_.nameprop -eq "Uscm0901P8"};")[0].click()
while ($ie.busy -or $ie.readystate -ne 4){ Start-Sleep -Milliseconds 200 }
do {
    start-sleep -m 200
    $ie = $shell.Windows() | where {$_.name -eq "Internet Explorer" -and $_.LocationName -eq "名古屋産業大学 UniVision"}
} while($ie -eq $null)

($doc.frames.item([ref]3).document.IHTMLDocument3_getElementsByTagName("input") | where {$_.id -eq "job_kind2"}).click()

```

```

while ($ie.busy -or $ie.readystate -ne 4){ Start-Sleep -Milliseconds 200 }
do {
    start-sleep -m 200
    $ie = $shell.Windows() | where {$_.name -eq "Internet Explorer" -and $_.LocationName -eq "
名古屋産業大学 UniVision"}
}while($ie -eq $null)

$w_in=$ie.document.frames.item([ref]3).document.frames.item([ref]0).document.IHTMLDocume
nt3_getElementsByTagName("input")

#発行番号あり
($w_in | where {$_.id -eq "issueNumFlg_1"}).click()

#履修中を出力しない
($w_in | where {$_.id -eq "rishuOutFlg_1"}).click()

#発行日付
#($w_in | where {$_.name -eq "issuedDate_yy1"}).value="2019"
#($w_in | where {$_.name -eq "issuedDate_MM1"}).value="11"
#($w_in | where {$_.name -eq "issuedDate_dd1"}).value="02"

#学籍番号
($w_in | where {$_.name -eq "studentNum"})[0].value=$stu_id

#実行ボタンのダイアログ確認を無効化
$jsCommand = "function confirm() { return true; }"
$document = $ie.document.frames.item([ref]3).document.frames.item([ref]0).document
$window = $document.parentWindow
$window.execScript($jsCommand, 'javascript') | Out-Null

($w_in | where {$_.value -eq "実行"}).click()
while ($ie.busy -or $ie.readystate -ne 4){ Start-Sleep -Milliseconds 200 }

do {
    $ie = $shell.Windows() | where {$_.name -eq "Internet Explorer" -and $_.LocationName -eq "
名古屋産業大学 UniVision"}
}while($ie -eq $null)

$w_in=$ie.document.frames.item([ref]3).document.frames.item([ref]0).document.IHTMLDocume
nt3_getElementsByTagName("input")
($w_in | where {$_.value -eq "実行結果一覧画面へ"}).click()
while ($ie.busy -or $ie.readystate -ne 4){ Start-Sleep -Milliseconds 200 }

do {
    $ie = $shell.Windows() | where {$_.name -eq "Internet Explorer" -and $_.LocationName -eq "
名古屋産業大学 UniVision"}
}while($ie -eq $null)

$exc=($doc.frames.item([ref]3).document.IHTMLDocument3_getElementsByTagName("td") |
where {$_.innerText -match "実行中.*|起動待.*" })
$elm=($doc.frames.item([ref]3).document.IHTMLDocument3_getElementsByTagName("img")
| where {$_.nameProp -eq "download.png" })
$log=($doc.frames.item([ref]3).document.IHTMLDocument3_getElementsByTagName("img") |
where {$_.nameProp -eq "log.png" })

```

```
while ($exc -ne $null) {
    ($doc.frames.item([ref]3).document.IHTMLDocument3_getElementsByTagName("input") |
    where {$_.value -eq "再表示"}).click()
    while ($ie.busy -or $ie.readystate -ne 4){ Start-Sleep -Milliseconds 2 }

    $selm=($doc.frames.item([ref]3).document.IHTMLDocument3_getElementsByTagName("img") |
    where {$_.nameProp -eq "download.png" })

    $exc=($doc.frames.item([ref]3).document.IHTMLDocument3_getElementsByTagName("td") |
    where {$_.innerText -match "実行中*" })
}
if (($selm.Length -eq 0) -or ($selm.Length -eq $null)) {
    $selm.click()
} else {
    $selm[0].click()
}

while ($ie.busy -or $ie.readystate -ne 4){ Start-Sleep -Milliseconds 200 }
do {
    start-sleep -s 1
    $ie = $shell.Windows() | where {$_.name -eq "Internet Explorer" -and $_.LocationName -like
    "成績証明書印刷*" } | Select-Object -First 1
} while ($ie -eq $null)

while ($ie.busy -or $ie.readystate -ne 4){ Start-Sleep -Milliseconds 200 }

#ウインドウをアクティブ化
Add-Type -AssemblyName Microsoft.VisualBasic
Add-Type -AssemblyName System.Windows.Forms
$window_process = Get-Process -Name "iexplore" | ? {$_.MainWindowHandle -eq $ie.HWND}
[Microsoft.VisualBasic.Interaction]::AppActivate($window_process.ID) | Out-Null
Start-Sleep -s 5
[Microsoft.VisualBasic.Interaction]::AppActivate($window_process.ID) | Out-Null
Start-Sleep -Milliseconds 1500
[System.Windows.Forms.SendKeys]::SendWait("%p") ; Start-Sleep -Milliseconds 500
Start-Sleep -Milliseconds 2000
[System.Windows.Forms.SendKeys]::SendWait("{TAB}") ; Start-Sleep -Milliseconds 300
[System.Windows.Forms.SendKeys]::SendWait("{TAB}") ; Start-Sleep -Milliseconds 300
[System.Windows.Forms.SendKeys]::SendWait("{TAB}") ; Start-Sleep -Milliseconds 300
[System.Windows.Forms.SendKeys]::SendWait($w_maisu) ; Start-Sleep -Milliseconds 300
[System.Windows.Forms.SendKeys]::SendWait("{TAB}") ; Start-Sleep -Milliseconds 300
[System.Windows.Forms.SendKeys]::SendWait("{ENTER}")
```

【参考文献】

- [1] 厚生労働省、「働き方改革を推進するための関係法律の整備に関する法律」について
https://www.mhlw.go.jp/stf/seisakunitsuite/bunya/0000148322_00001.html
(2019年11月1日 閲覧)。
- [2] UiPath 株式会社 Web ページ、<https://www.uipath.com/ja/> (2019年11月1日 閲覧)。
- [3] Winactor Web ページ、<https://winactor.com/> (2019年11月1日 閲覧)。
- [4] Vector UWSC Web ページ、<https://www.vector.co.jp/soft/winnt/util/se115105.html>
(2019年11月1日 閲覧)。
- [5] 清水 理史、『できる UiPath 実践 RPA』、株式会社インプレス、2019。