

Using Native Citrix in UiPath Automation for Citrix Applications

Himadeep Movva*

Citation: Movva H. Using Native Citrix in UiPath Automation for Citrix Applications. *J Artif Intell Mach Learn & Data Sci* 2022, 1(1), 2385-2389. DOI: doi.org/10.51219/JAIMLD/himadeep-movva/515

Received: 03 March, 2022; **Accepted:** 28 March, 2022; **Published:** 30 March, 2022

*Corresponding author: Himadeep Movva, Independent Researcher, USA, E-mail: movvahimadeep@gmail.com

Copyright: © 2022 Movva H., This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

ABSTRACT

The challenges corresponding to automating applications that require OCR/Image-based automation are prone to errors and frequent application exceptions, mainly because applications with available selectors have not been loaded while Bot performs the operations. The reasons could be many, including but not limited to screen resolution, bandwidth and slow processing for text-heavy screens. Justifiably, the percentage of successful transactions remained lower, sometimes even under 50%, based on the application type and stability. There's a pressing need to address these issues and expand the solution of UiPath, especially for thick client applications such as Citrix apps and Remote desktop applications. native Citrix in UiPath is a significant step forward towards this effect. This paper explores the functionality of Native Citrix in UiPath and explores pathbreaking problems that it has solved.

Keywords: Native Citrix, ICA, OCR, UiPath remote run time, Citrix extension, UiPath and invoice processing

1. Introduction

Integrating Native Citrix in UiPath enhances its automation features through direct interaction, using UI elements of Citrix and remote desktop applications. This state-of-the-art feature also eliminates the need to use image-based automation – a feature that comparatively improves efficiency. As native Citrix uses UI elements to interact with the application, it increases accuracy and efficiency. It operates without the need for any extra settings except Citrix plug-in the client machine and remote run time in the server machine. In this research paper, the need for native Citrix and the problems that it resolved is discussed. Also, infrastructure requirements, installation steps, usage of components and communication channels of native Citrix is mentioned.

2. Need for Native Citrix

There are two types of applications: thin client and thick client applications. Thin client applications are the ones that have

available UI elements for the selectors to be fetched reliably. Robust UI automation can be configured for processes that involve thin client applications. However, a few applications, such as Citrix or remote desktop applications, may not have immediate selectors, requiring the UiPath to configure image-based activities as the window will be just like an image. These types of applications are called thick client applications. In most cases, an interface or icon of the application in the Citrix server will be made available on the remote desktop server so that, based on UI actions, commands will be sent and run on the remote server.

Maintaining image-based automation is difficult and prone to frequent errors, making it resource-intensive. Click-based UI interaction is reliable than click based on image coordinates. Through selectors, native Citrix facilitates UI interaction on applications hosted on Citrix, preventing resolution changes and, hence, reducing error percentage in the live environment. Additionally, error checks can be reliably done using selectors

and portability across different environments can be achieved. Also, another challenge that native Citrix resolved is that image-based activities require an application to be available in the foreground, while using native Citrix, execution can be performed in the background¹.

Also, any current UiPath automation that uses OCR or image-based techniques can be revamped and redesigned for native Citrix in UiPath. Using click image, click OCR text, anchor base activity and various other image-based activities will often result in unpredictable efficiency, sometimes even low efficiency. This is because image-based activities are less reliable and the performance of those types of activities depends a lot on how the screen is loaded and with what bandwidth. If there is a slight reduction in the internet bandwidth, causing the bot to be unable to identify screens properly, the bot might fail the current transaction. Also, much trial and error is required while configuring image-based activities. Revamping the design with native Citrix, wherever applicable, makes the bot rely on UI automation as the selectors are retrieved effectively through communication between remote run time in the server and Citrix extension on the client. This solution would be a game changer for any pdf-heavy automation in supply chain, finance, human resources and legal. Especially in accounts payable, accounts receivable and procurement, using native Citrix will significantly increase the efficiency of the processes. This solution can be applied to almost any Citrix app with an icon on the machine to open. Still, processing happens at the server level instead of the process happening at the local level.

UiPath supports interactive UI selection with high DPI (Dots per inch). If the DPI setting in the machine is changed, a Citrix session needs to be in place for the changes to be in effect. If not, there would be issues in reliably identifying UI elements. Utmost precaution must be taken while using multiple monitors and automating Citrix applications as element selection and, ultimately, detection would be affected by resizing the applications in Citrix. If DPI settings are incorrectly handled, selectors may be misaligned or fail to work, leading to automation failures. Proper DPI configuration ensures stable and precise automation across different screen resolutions. The Citrix Extension cannot interface with the Citrix client (Citrix Receiver or Citrix Workspace) when deployed per computer. Before installing the Citrix UiPath Extension, install the Citrix client on each computer. The related Citrix session goes into an idle state and disconnects if a Citrix App is not used for a period of time. The Citrix application server sets the idle disconnect timeout setting, which is typically around 30 minutes. As with a normal user, the Click and Type Into activities automatically send hardware events to the Citrix App. By doing this, the Citrix App is kept from going into an idle state. However, the Citrix App may go into idle mode and display the following message if you utilize background automation exclusively for longer than 30 minutes (for example, click action with the SendWindowMessages feature enabled)²:

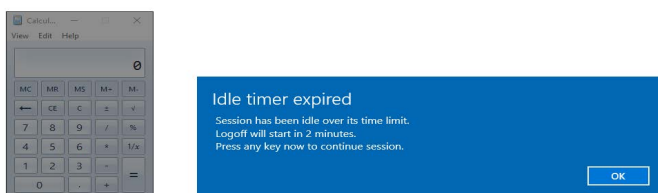


Figure 1: Idle timer expired window.

2.1. Components of native Citrix

2.1.1. UiPath extension for Citrix (Client-side component):

Once enabled, this extension in UiPath Studio lets the bot communicate with Citrix applications by enabling the generation of native selectors for UI elements. This extension enables direct interaction with UI elements, eliminating the need for image-based automation.

Steps to Install Citrix Extension are provided below

Home->Tools->Extensions->select Citrix->Install

This extension establishes a connection between the UiPath robot and the Citrix session using the ICA (write more info on this) protocol. The Citrix extension sends commands to the Citrix environment through the UiPath Remote Run time installed on the Citrix server.

2.1.2. UiPath remote run time (Server-side component):

This component is installed on the Citrix server and enables the communication between the Citrix environment and UiPath Citrix extension, facilitating native Citrix automation. UiPathRemoteRuntime.msi installer is installed on the Citrix server. This component resolves key challenges that were posed by previously used image-based or OCR-based automation by enabling interaction with UI elements and selectors in Citrix applications. The remote run time opens a virtual communication channel with the UiPath extension and the Citrix receiver. This functionality is leveraged when the bot interacts with an element, enabling runtime to fetch selector details of the element and send them back to the client.

2.2. Why computer vision can't be used instead of native Citrix

Computer vision identifies elements based on images instead of UI elements. So, identification is based more on appearance than underlying properties, making the CV prone to errors in case of frequent changes in layouts. When Native Citrix is installed and remote run time is configured, Citrix automation interacts directly with the application's controls, bypassing the need for image-based controls. Also, CV use can be resource-intensive, leading to more costs and slower processing times. Further, CV performance is affected by screen resolution and font changes, making it less reliable in such scenarios where UI elements can be accessed through a communication channel set up between native Citrix and Citrix remote run time. CV workflows often require transmitting screen images to external servers (ex., UiPath's CV Local server). This aspect is very tricky from a security aspect, especially regarding sensitive data in healthcare or finance. In contrast, Native Citrix adheres to and complies with security, offering built-in security protocols. It is easier to enforce based on encryption or role-based access due to its non-reliance on image-based data. CV identifies elements based on visual patterns and cannot interpret functional context. When there is a need to click on any drop-down option that is not immediately visible and will only be visible after a keyboard shortcut, native Citrix can accomplish that task while CV cannot.

Furthermore, Computer vision uses AI models to recognize elements - a feature that may introduce false positives or negatives. Native Citrix doesn't require such models and is more deterministic in capturing and identifying UI elements. Furthermore, Native Citrix uses Citrix Virtual Channel, a dedicated channel between UiPath and Citrix servers, making

it more efficient and reliable. In summary, Native Citrix is more feasible in environments where selectors can be accessed through a channel established through the Citrix plugin. While computer vision is a fallback option for legacy systems that lack API, native Citrix is far better for enterprise-grade automation due to better reliability, efficiency, scalability and security features.

2.3. How native Citrix automation works and the need for it:

To increase automation capabilities, it is possible to use UI activities and wizards with Citrix Apps. By installing the extension for Citrix on the client machine and UiPath remote runtime on Citrix Virtual Apps application servers, automation in Citrix apps can be created as if they were local applications. As selectors are natively generated for Citrix app elements, we can utilize the Ui Automations package and activities within the package fully. UiPath makes direct access to Citrix user interface elements rather than depending on image recognition. This makes it possible to identify elements with selectors, dynamically automate keyboard and mouse movements and increase automation dependability. It uses a Virtual Channel for Communication. The UiPath Citrix Extension and Remote Runtime create a virtual channel. The local UiPath robot and the Citrix-hosted applications can communicate in both directions via this channel.

Before discussing the need for native Citrix, this paper delves deep into why traditional approaches might not work better. Before native Citrix, RPA tools depended on indirect methods to interface with Citrix applications. The idea behind image-based automation is to take pictures of user interface elements and then act on them based on how they look. Despite being popular, this approach has several drawbacks:

2.4. Image-based automation

- **Low accuracy:** Exactly matching an image depends on various factors, including screen resolution, bandwidth and color contrast, among others. This makes automation error-prone and non-fail-proof.
- **Reliance on screen:** In case of a change in resolution or display settings, automation may fail
- **Transaction times:** Image matching and recognition take longer and may pose challenges when automation must be SLA-dependent.
- **Complexity in debugging:** Even a small variation or change in UI may cause automation to break, requiring intermittent efforts during maintenance.

2.5. OCR-based automation

- **Low accuracy:** OCR finds it hard to capture content when there are inconsistent layouts, fonts or blurred text
- **Processing time:** Time to process a transaction is increased, straining resources and affecting SLA and timelines
- **Functionality drawbacks:** While OCR can read text, it cannot interact with UI elements directly.

2.6. Keystroke & hotkey automation

While some automation may rely just on keystrokes to navigate to applications, this approach is not reliable due to the following reasons

- **Zero UI interaction:** The bot does keystrokes ex: up, down, left, tab, etc. and there is no way to check if the bot is in the right place and doing the right update
- **Error prone:** If any unexpected popup or notification appears, the automation may fail
- **Lack of scalability:** It is as difficult to maintain complex hard-coded keystroke sequences as it is to develop.

Native Citrix automation offers a reliable way around these restrictions by enabling direct interaction between UiPath and UI elements within Citrix applications. Thanks to the installation of UiPath Remote Runtime on the Citrix server, the UiPath Robot and the Citrix-hosted applications can communicate directly. Native Citrix automation, as opposed to image-based automation, permits direct interaction with user interface elements, enabling RPA bots to: Dynamically identify items regardless of scaling or screen resolution, work with buttons, text fields, dropdown menus and other UI components, use dependable selectors; and avoid OCR errors while retrieving structured data. Because automation workflows don't depend on image processing or OCR, they operate much more quickly³. Native Citrix facilitates UI interaction through selectors, reducing processing time and resulting in faster execution, lesser resource usage and higher efficiency and scalability for automated operations with large volumes.

Required Infrastructure for Native Citrix:

Extension for Citrix

Installing the Extension for Citrix makes it possible to automate Citrix Apps natively. It can be installed from Studio or the Command Prompt

Installation from Studio

Below are the steps to install the Citrix extension from the studio

- Open Studio
- Navigate to Home->Tools-> UiPath Extensions
- Select Citrix->Install

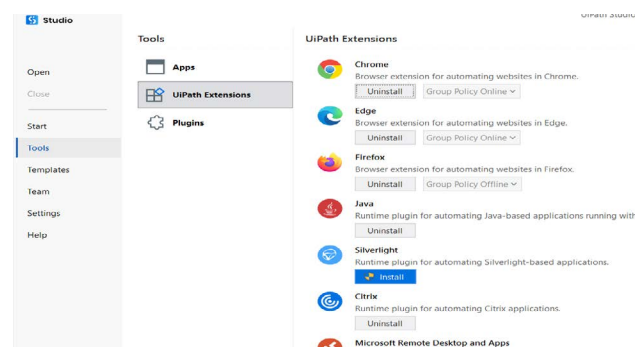


Figure 2: Install the Citrix runtime plugin.

Restart all Citrix applications for the changes to take effect. The UiPath Extension for Citrix is now installed in the Citrix install directory.

2.7. Installation from command prompt

Click the Windows Start button and type cmd in the search field.

Right-click on Command Prompt and run it as administrator.

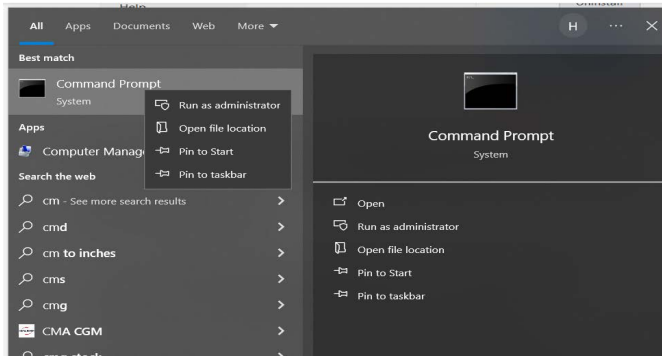


Figure 3: Run as Administrator.

Change the directory to the **UiPath** installation folder (cd C:\Program Files\UiPath\Studio\UiPath for per-machine installations, cd %localappdata%\Programs\UiPath\Studio\UiPath for per-user installations).

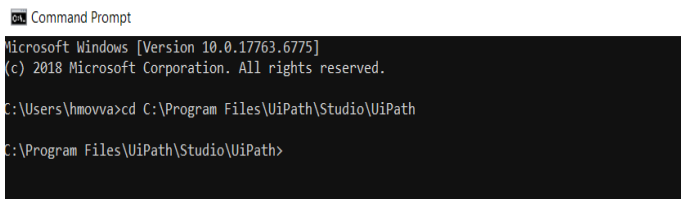


Figure 4: Change the directory in the command prompt.

Use the SetupExtensions /CitrixClient command to install the extension. A confirmation dialog box is displayed

Restart all your Citrix applications for the changes to take effect. The UiPath Extension for Citrix has been installed.

2.8. UiPath remote runtime

UiPath Remote Runtime makes it easier for a remote desktop or application, like Citrix Virtual Apps, to communicate with a specific UiPath extension, such as the UiPath Extension for Citrix, the UiPath Extension for Windows Remote Desktop and Apps or the UiPath Extension for VMware Horizon. To generate selectors natively in UI Explorer, it collects information about remote apps’ desired user interface elements and forwards it to the appropriate extension⁴. Starting with Citrix Virtual Apps and Desktops 7 2109, all custom virtual channels are blocked by default, preventing UiPath Remote Runtime from working correctly. To solve this issue, the UiPath virtual channel must be added to the allow list policy. The remote runtime component comes with a Citrix-ready certification, making it a trusted solution for working with Citrix technologies.

Depending on the type of environment you want to build your automation projects on, the UiPath Remote Runtime component can be installed as follows:

On Citrix Application servers - communicates with the UiPath Extension for Citrix⁵.

On VMware Horizon virtual machines - communicates with the UiPath Extension for VMware Horizon⁶.

On Windows Remote Desktop machines - communicates with the UiPath Extension for Windows Remote Desktop and Apps⁷.

Please note that the corresponding UiPath extensions must be installed on the client machine.

The UiPathRemoteRuntime.msi installer can be downloaded from:

Product Downloads in the UiPath Customer Portal

Resource Center in the UiPath Automation Cloud. To access Resource Center, log in to your Automation Cloud Organization and click the Help button on the navigation bar.

Once the UiPath Remote Runtime component is installed, it registers a task in the Task Scheduler so that the UiPathRemoteRuntime.exe process is started at every user login. The UiPath Remote Runtime component is necessary to connect an application or desktop server to a matching UiPath extension installed on a client computer. This eliminates the need for OCR and image recognition tasks by generating selectors directly on the client computer where Studio is installed.

2.9. Remote runtime architecture

The Robot uses an RPC channel to communicate the required commands to the Citrix or Windows Remote Desktop extension when a process is launched. The version of the UI Automation package that was used to create the process is also described in this information package. The UiPath Remote Runtime component, located on the Citrix Application servers or Remote Desktop computer, receives the information package from the relevant extension via an ICA virtual channel. The UiPath Remote Runtime component tells the Robot Executor which driver to use based on the version of the UI Automation package used to build the process.

2.10. Multiple ICA channels

The Remote Runtime Component and the corresponding extension are communicated via an internally developed ICA channel. Multiple ICA channels can be used, but the Citrix Receiver, for instance, already uses several channels for the clipboard, audio and more.

2.11. Data transfer and bandwidth considerations

Once the Remote Runtime component is installed, automation projects can be created without creating any firewall rules.

Data transfer between the Remote Runtime and extension requires little bandwidth usage. For example, each action the Robot performs (such as a Click activity) transfers around 4 KB of data

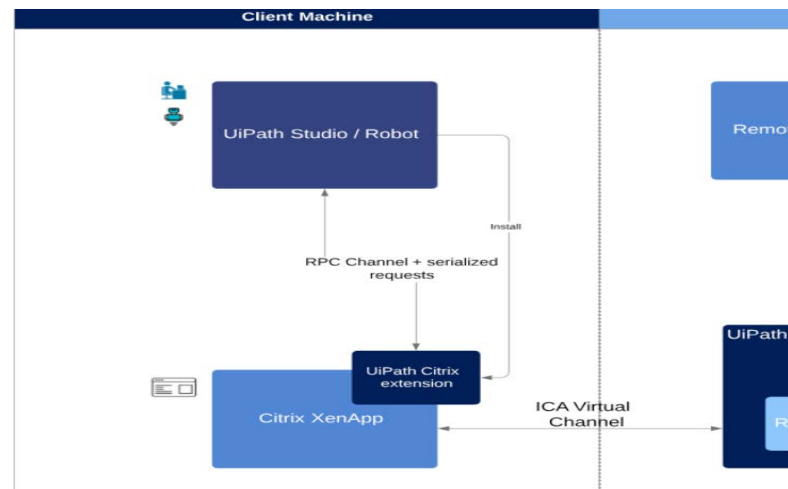


Figure 5: The architecture of native Citrix in UiPath.

The UiPath Remote Runtime is installed on the Citrix application server, while the UiPath Citrix Extension is on the client machine. Communication between these components is done through an ICA virtual channel. This is the same mechanism used for transmitting data such as audio and video for Citrix Apps, so no additional configuration is needed.

UiPath enabled the use of UI activities and wizards with Citrix Apps to expand automation capabilities. We can now construct automation projects for Citrix Apps as if they were local applications by installing the UiPath Remote Runtime component on the Citrix Virtual Apps (formerly known as Citrix XenApp) application servers and the Extension for Citrix on the client computer. We can use the entire set of UI Automation activities (including Click, Type Into, Get Text, Extract Data and more) and mouse and keyboard event monitoring activities with selectors natively produced for Citrix App elements.

2.12. Selectors for Citrix apps

After completing the configuration steps, UiPath can detect individual UI elements from a Citrix App and get selectors for them⁸.

This is what the generated selector for Calculator opened as a Citrix App looks like:

```
<wnd app='win32calc.exe' cls='CalcFrame' isremoteapp='1'
title='Calculator' />
<wnd ctrlid='137' />
```

This is what the generated selector for Calculator opened as a local application looks like:

```
<wnd app='win32calc.exe' cls='CalcFrame' title='Calculator'
/>
<wnd ctrlid='137' />
```

2.13. Important distinctions

`isremoteapp='1'`: This property shows that the program operates in a remote virtualized environment, such as Windows Virtual Desktop or Citrix. The window (`wnd`) and control (`ctrlid='137'`) are identified by the remaining selector structure. The only difference between the two selectors is the `isRemoteApp='1'` attribute. This gets added automatically when selecting a UI element from a Citrix App. Aside from the `'isremoteapp'` attribute, there is no distinction between components of the UiPath process in the Citrix app and in other systems. Most of the activities in UI automation, such as click, type into, get text, find element and get attribute, can be used for native Citrix.

UiPath must behave differently when an application is running in Citrix because

- The program is streamed from a Citrix server rather than being installed locally.
- As direct UI element interaction is limited, alternative automation techniques are needed.
- In order for UiPath to decide whether to use native Citrix automation or image-based/OCR-based automation, the selector must identify it as a remote application (`isremoteapp='1'`).

3. Conclusion

Citrix Automation in UiPath provides a reliable and effective way to automate tasks using UI interactions instead of using image based activities. Businesses may improve productivity, decrease manual errors and streamline operations using UiPath's advanced features. One major benefit is the ability to interact with virtual apps, just like interaction with local ones a benefit, which makes managing and automating complicated operations simpler. Also, using Native Citrix instead of image-based automation makes the automation deliver high accuracy and efficiency comparatively. Justifiably, RPA implementation in virtual settings is revolutionized by native Citrix automation in UiPath. As there is direct interaction with UI elements, sometimes, even if the screens are dynamic, it can be handled using native Citrix. If the previous automation was developed using image-based activities, the workflows can be revamped and reconfigured using native Citrix for better efficiency and stability.

4. References

1. <https://forum.uipath.com/t/difference-between-image-recording-and-native-citrix-automation/397850/2>
2. <https://docs.uipath.com/studio/standalone/2021.10/user-guide/known-issues-and-limitations-citrix>
3. Kauppinen M. "Robotic Process Automation in containerized environment," theseus, 2020: 34.
4. <https://docs.uipath.com/studio/standalone/2021.10/user-guide/about-uipath-remote-runtime>
5. <https://docs.uipath.com/studio/standalone/2021.10/user-guide/extension-for-citrix>
6. <https://docs.uipath.com/studio/standalone/2021.10/user-guide/extension-for-vmware-horizon>
7. <https://docs.uipath.com/studio/standalone/2021.10/user-guide/extension-for-windows-remote-desktop>
8. <https://docs.uipath.com/studio/standalone/2021.10/user-guide/about-automating-citrix-technologies>