



Optimizing journey  
time to robotic  
process automation



Leading enterprises across the world are willing and keen to adopt Robotic Process Automation (RPA). However, they encounter many roadblocks in their journey. More often, three major challenges affect effective implementation of RPA in an enterprise:

- Fallouts of automation journey, specifically 'Automatable Exception'
- Manual scheduling leading to lower bot utilization
- Absence of alert mechanism to prevent bot idle time and bot code defects

This has resulted in the need to optimize some of the steps of RPA journey in order to ensure efficient implementation and faster outcomes.

### Hurdles in the RPA journey

In many organizations, most processes have exceptions that are not pre-identified and well documented. These organizations also may not have detailed standard operating procedures (SOPs) of their processes and sub processes drawn at a keystroke level. Therefore, they fail to provide relevant documentation to the bot developers. This leads to incomplete design of the code where exception scenarios are not captured – leading to bot errors. Such fallouts tend to be very high, which in turn compels the bot governance team to ask for accurate process maps of the exception scenarios. The governance team is entrusted with the task of increasing the overall bot utilization, process amenability and efficiency on a continuous basis.

Lack of detailed low-level SOPs requires RPA consultants to draw keystroke maps of every process step. The objective is to capture all the steps of the process along with respective timestamps. Then, these keystroke maps are manually assessed for possibility of automation. In most cases, the system and workflows have fields containing both structured and

unstructured data; the task of identifying the data type at each step is also a manual process.

Once the process is automated, the task of allocating or deallocating runtime bots is assigned to agents known as bot runners. The task of bot configuration is manual. The run time of the bot is constrained by the availability of human agents and unavailability of bot runners. Therefore, when the bot finishes processing the transactions in one queue, it remains idle until the human agent manually switches the bot over to another queue. Thus, there is a need for a controller engine to bridge this gap by identifying the available or free bot and scheduling them to manage work queues automatically.

Lack of auto allocation and auto de-allocation of bots across queues compels businesses to either create custom code for a controller engine, or assign the responsibility to human agents for running bots as per volumes in the workflow.

Typically, as human agents are entrusted with multiple tasks, they do not sit in front of the console to monitor the performance of the runtime bots continuously. Therefore, when the bot code is defective, the runtime of the bots becomes skewed significantly. The buggy bot may end up in an infinite loop, or its faulty memory management design might cause the system to freeze. If there are no auto-identification of such faulty situations with auto-email alert mechanism, the bots remain silent. However, a trigger – like an email, SMS or tweet – alerting the human agent to reboot the bot machine immediately helps address the memory freeze. Otherwise, bot utilization suffers even though the bot is readily available for reuse.

### Agile route to automation

In order to accelerate process automation, the traditional RPA deployment approach has four phases:

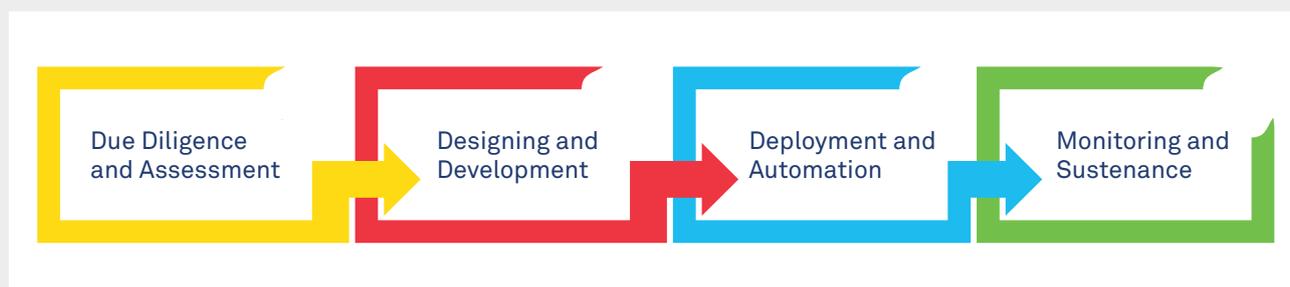


Figure 1: Four Phases of RPA Life Cycle

To ensure better results, there is a need to introduce automation activities in the first and last phases, i.e. Due Diligence and Assessment Phase and Monitor & Sustenance Phase

## Optimal Process Designer Bot will speed up the evaluation of RPA feasibility across enterprise processes.

### Due Diligence and Assessment Phase with Optimal Process Designer Bot

In order to accelerate process mapping, bot developers need to create an Optimal Process Designer Bot. The objective of this tool is to identify the fields and forms of application that are invoked during process execution, and record the length of time spent across the fields during data entry. It also identifies data manipulation through timestamp mapping, as well as areas

where structured data and free flowing data are involved. These details are captured in data logs. Secondly, developers need to integrate the data-log creation with simultaneous image capture. Together with timestamp and image of the process step taken synchronously, you get the keystroke map of the process, which needs to be presented in a tabular form for the evaluation of the automation percentage across the steps. The evaluation process requires manual intervention to identify the steps containing structured and unstructured data. This Optimal Process Designer Bot will speed up the evaluation of RPA feasibility across enterprise processes.

### Monitor and Sustenance Phase with Control Bot

For creating a controller bot, developers need to know the applications and queues that must be monitored. They should be able to access queue details for pending volumes and in-processing volumes, identify the count of allocated bots, and free bots at any given instant, and estimate the standard AHT of transactions that are being auto-processed through bots. Developers should receive the priority schedule of the queues, scheduled down times, holiday list, and the escalation matrix. A Controller Engine should run as a service in the background and needs to be tasked with the following set of activities:

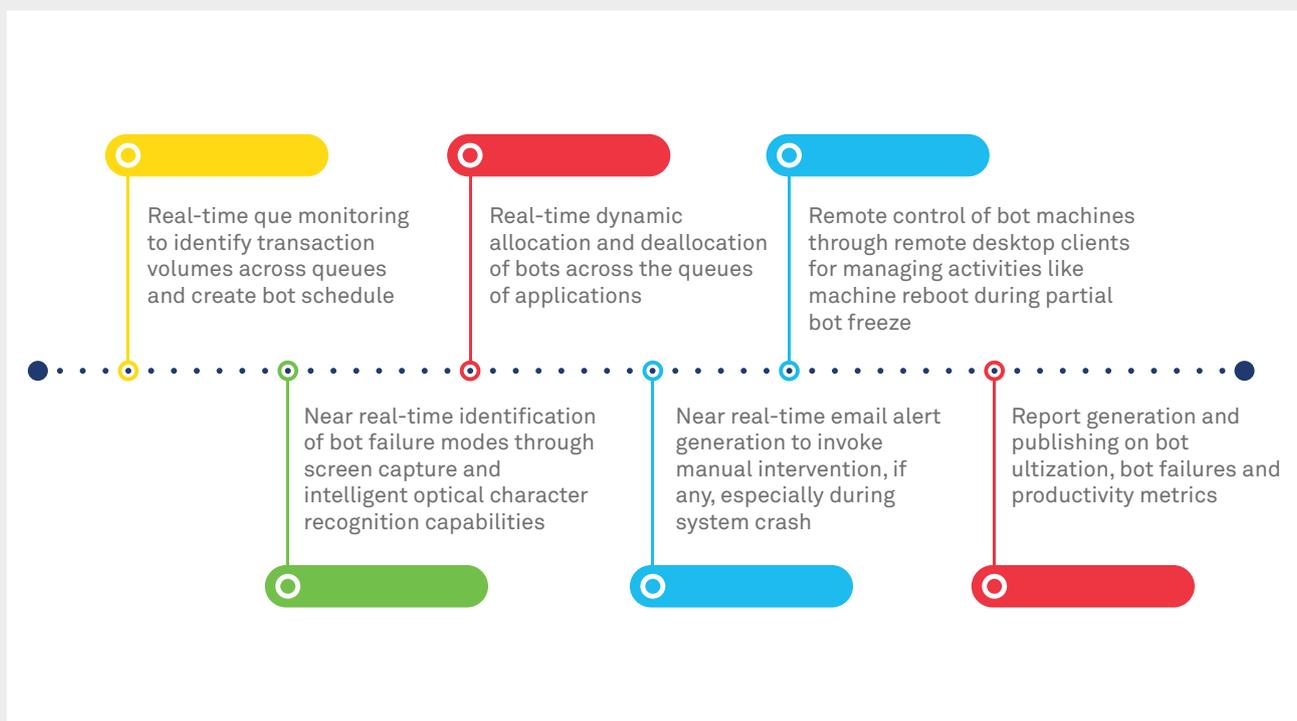


Figure 2: Responsibilities of Control Bot

## Advantage automation

Through automated data capture tool, keystroke maps of all types of processing across all users can be captured over time, and this map can be developed on-the-go. This will create true traceability of every type of processing, both STP and Exceptions, in an automated manner. These maps will be the best tool for error detection, error prevention, and improve accuracy, reduce

time for learning curve and generate more ROI – apart from its role in process automation (See Figure 3). Through Control Bot automation, we can optimize the usage of run-time bot licenses required for processing transactions with improved availability for other queues, thus reducing licensing costs, and eliminating dependency on manual user to schedule transactions.

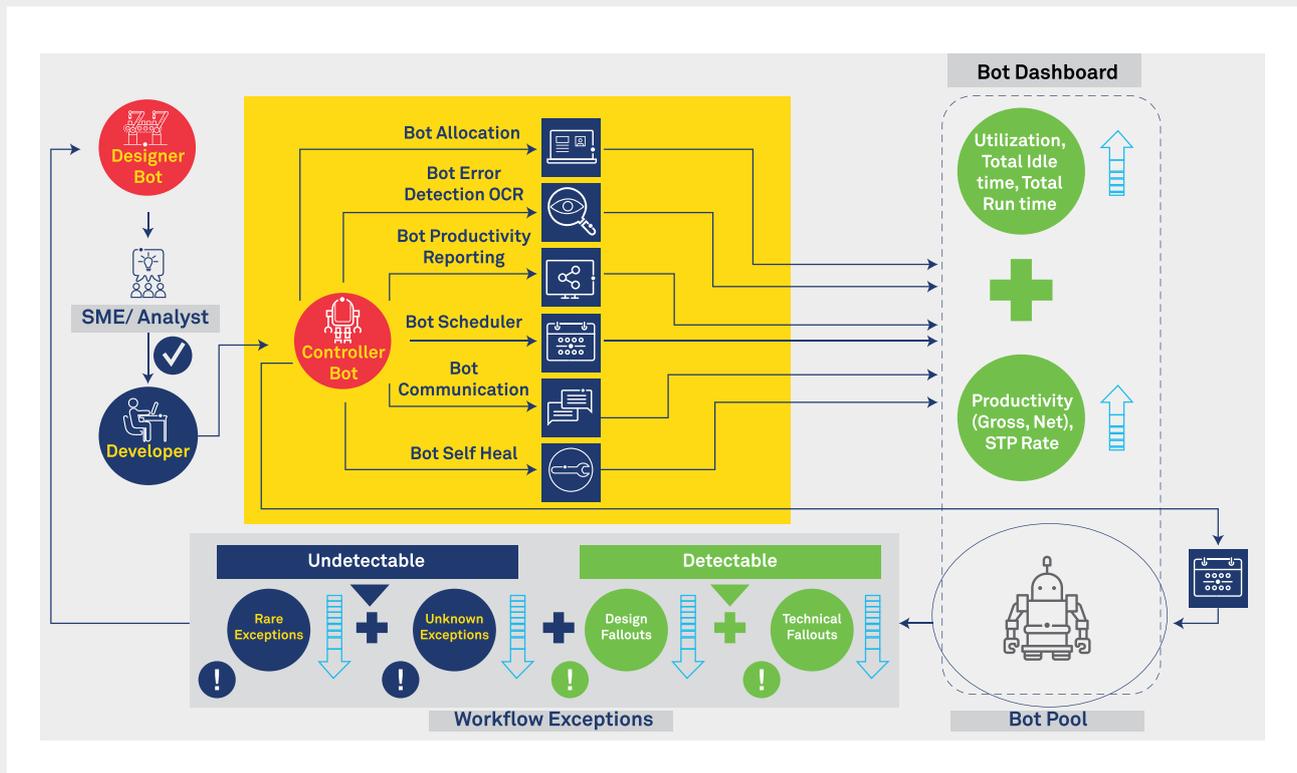


Figure 3: Automation optimization

## About the author

**Suprateem Moitra**  
Associate Vice President – Quality,  
Wipro Ltd

Suprateem has more than 15 years' experience improving process delivery through Robotic Process Automation and Artificial Intelligence for the Banking, Insurance, Healthcare, Supply Chain and KPO Industries.

Suprateem is a Quality Black Belt since 2015. He has a Masters in Management from the Indian Institute of Foreign Trade (IIFT) New Delhi and a Bachelor's in Engineering from the Indian Institute of Technology Kharagpur.

**Wipro Limited**

Doddakannelli, Sarjapur Road,  
Bangalore-560 035,  
India

Tel: +91 (80) 2844 0011

Fax: +91 (80) 2844 0256

**wipro.com**



Wipro Limited (NYSE: WIT, BSE: 507685, NSE: WIPRO) is a leading global information technology, consulting and business process services company. We harness the power of cognitive computing, hyper-automation, robotics, cloud, analytics and emerging technologies to help our clients adapt to the digital world and make them successful. A company recognized globally for its comprehensive portfolio of services, strong commitment to sustainability and good corporate citizenship, we have over 160,000 dedicated employees serving clients across six continents. Together, we discover ideas and connect the dots to build a better and a bold new future.

For more information,  
please write to us at  
**info@wipro.com**

