# Aspect Optimalization of Robotic Process Automation (Extended Abstract)

Petr Průcha
Faculty of Economics
Technical University of Liberec
Liberec, Czechia
petr.prucha@tul.cz

***Abstract*—Robotic Process Automation (RPA) is an expanding technology that is used around the world. RPA is used by many organizations for the automation of routine tasks. Due to the easy-to-use environment of RPA technology, it is easy to deploy new robotic processes in production. As an innovative technology, there is currently a lack of research and analysis to provide a full understanding of the performance of RPA robots. The proposed research investigates which methods lead to better efficiency, easier maintenance, and crash-free operation.**

***Keywords—Robotic process automation, optimization of RPA, performance and efficiency of RPA***

## I. INTRODUCTION

Robotic process automation is a software robot that completes routine tasks of a deterministic process. RPA applies the user's front-end and works according to a predetermined scenario employing exactly the same user interface as any employee of a certain organization. [1]

One of the reasons why RPA is so popular is due to the non-invasiveness of RPA into current IT systems, as well as the possibility to automate the activities of legacy systems without the need to invest in back-end architecture. Last but not least, investing in RPA is more economical than developing legacy systems. [2], [3]

RPA technology is already very advanced. Most leading RPA software vendors can in all probability automate 90 to 95% of deterministic user activity.[4]

Most RPA manufacturers strive to create a user-friendly interface with a good user experience to simplify work. A characteristic feature of RPA technology is called "Low-Code". Low-Code programming uses a minimum of code written by the developer and tries to simplify program development by using a graphical user interface, whereas the developer mainly uses pre-prepared functions of the integrated development environment (IDE). RPA platforms mostly use the advantages of Low-Code programming and RPA platforms can also be called Low-code development platforms (LCDP) [5]. Very often, Low-Code software is associated with the "Drag and Drop" function. [6]

This leads to opening coding and automation to citizen developers, who could have a lack of skills and training [7]. The development of custom Low-Code programs simplifies development and makes it possible to automate more activities with Low-Code RPA than it would be to automate processes with traditional coding [1], [8]. These two facts lead to poorly performing RPA robots, which are easy to build but hard to maintain [9]. Performance and maintenance are significant research challenges of the current state of the art in the field of robotic process automation [2].

### A. RATIONALE

Theoretical research indicates that there is a research gap in the investigatation of which aspects of RPA are possible to optimize. Many companies build and deploy RPA robots yet do not optimize the performance of the robots afterwards due to a lack of knowledge and resources. RPA developers can later use the knowledge from future researchers and implement some aspects into the development process of RPA. These aspects can be proposed as guiding principles on how to build and design RPA robots. Due to the novelty and rapid growth of RPA, guiding principles are lacking for how to build more efficient robots that are easy to maintain and crash-free.

Investigation into these research questions mitigate the cost of maintenance, hypercare and RPA developers, decrease time spent on the development of RPA, and promote an environment in which robots can complete more tasks allowing companies to use resources for further automation with more advanced technologies.

## II. RESEARCH GOAL AND RESEARCH QUESTIONS

The main goal isto find which attributes of RPA robots can be optimized so that RPAs are more efficient, easy to maintain and crash-free due to any exceptions.

From the main goal, several research questions followthat should answer the research goal.

- RQ 1: *What are the vital aspects of RPA which should be optimized?*
- RQ 2: *What are the options for increasing the performance/efficiency of the RPA robots*
- RQ 3: *What methods and techniques will help to simplify the maintenance of RPA robots?*
- RQ 4: *Which methods will contribute to the possible exceptions of RPA robots?*

## III. METHODOLOGY

In the first step, we research RQ 1 to understand the field of RPA. We conduct a literature search of scientific studies and analyses of the current state of the art. In addition, in-depth interviews will be conducted with RPA developers, head of Center of Excellence (CoE), and RPA leads who have real development experience. Based on their knowledge and the literary research, we can better understand the problem with greater insight.

Based on the discovered findings, we prepare the experiments to answer the research questions. For every experiment, an appropriate statistical hypothesis will be formulated with a method to validate the results.

RQ 2: For running the experiment to investigate RQ 2, it will be necessary to prepare a neutral testing environment. We will try to prepare testing scenarios to reflect reality. For testing, the scenarios will be crucial to choose the proper applications. We already know some requirements: We need at least one application with API support and Graphical User Interface (GUI) in one application and an application with a login. The following requirements will be added to the testing scenarios. For more relevant results, we will run the experiments on the most commonly used RPA software, which is UiPath, Automation Anywhere and Blue Prism [10].

One hypothesis for testing RQ 3 is that API is more stable without change compared to GUI. In a certain period of time, we will be observing changes in the code of API and also in the code of GUI and after some time we will compare what is more stable. In the experiment we again choose an application with API and GUI. The relevant application can be social media or enterprise resource planning ERP programs. We will prepare a set of these applications and deploy RPA robots to perform tasks and after a certain period, we will compare results to discover whether robots performing tasks via API are more stable than robots performing tasks via GUI.

RQ 4: Exception handling of RPA robots is mostly solved just by restarting the process and if it is system exceptions, then it usually helps. Many system exceptions are caused by waiting for loading the certain application or web service. In this research project we will try to analyze RPA logs and try to extract patterns from data with a focus on exceptions caused by loading the application.

As for other ideas of how to improve and optimize RPA, it is possible to iterate a new round of literary research and in-depth interviews.

## IV. CURRENT STAGE OF THIS RESEARCH PROJECT

The first phase of this research project has been completed including the collection of information from experts by means of in-depth interviews and the literature analysis. The first experiment will be testing RQ 2 on how to increase performance. The hypothesis is that the RPA robot completes the task faster through API than GUI. This should decrease the throughput time, and the RPA robots will be able to handle more tasks. Also, the use of the API should help with the maintenance of RPA robots since one of the other hypotheses is that more changes are released on GUI than API, and the new versions of GUI can cause the robot to crash.

RQ3: We will run the experiment described in the methodology to test that API is more stable than GUI. Another interesting fact that has to be tested is that when we are building RPA robots, we should reuse parts of the robot for the same task. If the process has the same parts in at least a few steps, for example, the login to an application, it would be better to reuse the robots for the part with the login. The advantage of this is that if changes occur in any application and RPA robots crash, it would be necessary to re-code the RPA robot to fix the error. If RPA developers reuse the robot, they only have to make the repair on one place/one robot. It is similar to guidelines for programing (Do-not repeat yourself - DRY).

Exception handling is in the early stage of the research project. The research project aims to improve errors with system exceptions but we also research some improvements for business exceptions of RPA robots. A possible solution is based on an analysis of RPA robot logs. An analysis of UI logs can show patterns of why robots crash due to system exceptions. Machine learning algorithms will be used for the analysis of the logs to find the rules. It is possible to use algorithms like C4.5, RIPPER or PART. If these algorithms do not fulfill their purpose, it will be necessary to conduct a more intensive study of machine learning and AI algorithms for the automatic recognition of patterns in data.

Handling business exceptions is more difficult because it is usually caused by human error. A common error is not adding all alternatives of situations to the process design document (PDD) or not fulfilling the assignments from the PDD in the solution design document SDD or later in the development stage. [11] The problem of discovering all routine traces is typical for process mining. Finding all routine traces has already been solved. The greater challenge is to discover the candidates of routine traces for automation [12], [13].

## ACKNOWLEDGMENT

This research was sponsored by the Technical University of Liberec via the SGS project with evidence number SGS-2021-1033. This research was conducted with the help of Pointee. We want to thank all participants of the in-depth interviews for their time and help.

## REFERENCES

[1] L. P. Willcocks, M. Lacity, and A. Craig, 'The IT function and robotic process automation', London School of Economics and Political Science, LSE Library, Oct. 2015. [Online]. Available: https://ideas.repec.org/p/ehl/lserod/64519.html

[2] R. Syed *et al.*, 'Robotic Process Automation: Contemporary themes and challenges', *Comput. Ind.*, vol. 115, p. 103162, Feb. 2020, doi: 10.1016/j.compind.2019.103162.

[3] A. Asatiani and E. Penttinen, 'Turning robotic process automation into commercial success – Case OpusCapita', *J. Inf. Technol. Teach. Cases*, vol. 6, no. 2, pp. 67–74, Nov. 2016, doi: 10.1057/jittc.2016.5.

[4] X. Lhuer, 'The next acronym you need to know about: RPA (robotic process automation)', 2016.

[5] C. Silva, J. Vieira, J. C. Campos, R. Couto, and A. N. Ribeiro, 'Development and Validation of a Descriptive Cognitive Model for Predicting Usability Issues in a Low-Code Development Platform', *Hum. Factors J. Hum. Factors Ergon. Soc.*, May 2020.

[6] S. Agostinelli, A. Marrella, and M. Mecella, 'Research Challenges for Intelligent Robotic Process Automation', in *Business Process Management Workshops*, Cham, 2019, pp. 12–18.

[7] M. Oltrogge *et al.*, 'The Rise of the Citizen Developer: Assessing the Security Impact of Online App Generators', in *2018 IEEE Symposium on Security and Privacy (SP)*, San Francisco, CA, May 2018, pp. 634–647. doi: 10.1109/SP.2018.00005.

[8] S. Anagnoste, 'Robotic Automation Process - The next major revolution in terms of back office operations improvement', *Proc. Int. Conf. Bus. Excell.*, vol. 11, no. 1, pp. 676–686, Jul. 2017, doi: 10.1515/picbe-2017-0072.

[9] D. Kedziora and E. Penttinen, 'Governance models for robotic process automation: The case of Nordea Bank', *J. Inf. Technol. Teach. Cases*, vol. 11, no. 1, pp. 20–29, Jun. 2020.

[10] 'Robotic Process Automation (RPA) Software Reviews 2021 | Gartner Peer Insights', *Gartner* https://www.gartner.com/market/robotic-process-automation-software (accessed Aug. 19, 2021).

[11] J. Schuler and F. Gehring, 'Implementing Robust and Low-Maintenance Robotic Process Automation (RPA) Solutions in Large Organisations', *SSRN Electron. J.*, 2018, doi: 10.2139/ssrn.3298036.

[12] A. Bosco, A. Augusto, M. Dumas, M. La Rosa, and G. Fortino, "Discovering Automatable Routines From User Interaction Logs," in BPM'19, Forum track, 2019.

[13] V. Leno, A. Augusto, M. Dumas, M. La Rosa, F. Maggi, and A. Polyvyanyy, 'Identifying candidate routines for Robotic Process automation from unsegmented UI logs', *ArXiv200805782 Cs*, Aug. 2020.