AIRPA: An Architecture to Support the Execution and Maintenance of AI-Powered RPA Robots

A. Martínez-Rojas[1(✉)], J. Sánchez-Oliva[2], J. M. López-Carnicer[1], and A. Jiménez-Ramírez[1]

[1] Departamento de Lenguajes y Sistemas Informáticos, Escuela Técnica Superior de Ingeniería Informática, Avenida Reina Mercedes, s/n, 41012 Sevilla, Spain
{amrojas,ajramirez}@us.es, jose.lopez@iwt2.org

[2] Servinform, S.A. Parque Industrial PISA, Calle Manufactura, 5, Mairena del Aljarafe, 41927 Sevilla, Spain
jmsanchezo@servinform.es

Abstract. Robotic Process Automation (RPA) has quickly evolved from automating simple rule-based tasks. Nowadays, RPA is required to mimic more sophisticated human tasks, thus implying its combination with Artificial Intelligence (AI) technology, i.e., the so-called intelligent RPA. Putting together RPA with AI leads to a challenging scenario since (1) it involves professionals from both fields who typically have different skills and backgrounds, and (2) AI models tend to degrade over time which affects the performance of the overall solution. This paper describes the AIRPA project, which addresses these challenges by proposing a software architecture that enables (1) the abstraction of the robot development from the AI development and (2) the monitor, control, and maintain intelligent RPA developments to ensure its quality and performance over time. The project has been conducted in the Servinform context, a Spanish consultancy firm, and the proposed prototype has been validated with reality settings. The initial experiences yield promising results in reducing AHT (Average Handle Time) in processes where AIRPA deployed cognitive robots, which encourages exploring the support of intelligent RPA development.

Keywords: Robotic Process Automation · Artificial Intelligence · Industrial project

1 Introduction

The term Robotic Process Automation (RPA) refers to a software paradigm in which robots are programs that mimic the behavior of human workers interacting

This research has been supported by the NICO project (PID2019-105455GB-C31) of the Spanish Ministry of Science, Innovation and Universities and the AIRPA project (EXP00118029/IDI-20190524, P011-19/E09) of the Center for the Development of Industrial Technology (CDTI).

with information systems (IS) [3,13]. This paradigm has become increasingly popular because RPA is of great interest to organizations [5].

In this context, RPA solutions based on Artificial Intelligence (AI) – called intelligent RPA solutions – are receiving increasing attention, as the combination of both disciplines offers and several advantages [2]. On the one hand, AI methods enhance RPA solutions by providing new capabilities that enable a more significant number of end-to-end processes to be automated. On the other hand, RPA solutions produce data on the execution of the process themselves, allowing periodic training of the AI models, leading to continuous improvement of the model metrics [9].

The use of this kind of component involves different challenges when a methodology, architecture o role specification proposal does not exist.

First, a data scientist (i.e., a professional in charge of processing structured data to extract relevant information from it) is required to develop the cognitive components and, without an abstraction role, needs to know the business process to configure a model for each business case. This fact leads to a strong dependency between the data scientist and the RPA developer role (i.e., professional in charge of designing and developing software robots) who must know the business process to automate it[1]. Secondly, the performance of these components in a production environment depends on the data model performance, which tends to degrade over time [6]. Degradation refers to multiple reasons, such as: (1) the evolution of the business over time and the AI obsolescence in the new business context, (2) the AI technology advance caused by the new scientist research in the AI, that can improve the AI performance and accuracy, led previous AI models obsolete, and (3) the need to re-training models to increase their accuracy and performance for a specific task. This problem arises the need to conduct the AIRPA project, a platform with an architecture that allows solving the challenges encountered, (1) separate the work of RPA developer and data scientist to abstract robot construction from model development, and (2) control, monitor, and support the robots to ensure quality maintenance of the intelligent RPA components.

As shown in Fig. 1, RPA developers are in charge of building the robot that automates the process. However, such developers may lack skills related to AI. For this, AI services (e.g., text-to-image, speech-to-text recognition, sentiment analysis, image anomaly detection, and others) have to be black boxes that always maintain an acceptable level of accuracy in their responses. The data scientist's implementation of these cognitive solutions makes their use transparent to the RPA developer. To these needs, the AIRPA project provides an architecture that supports the abstraction between both roles, and continuous monitoring mechanisms to ensure the quality of AI models, for both new release deployments and retraining.

The rest of the paper is organized as follows. Section 2 describes the project context and a set of example cases. Sect. 3 presents the AIRPA project. Section 4

[1] https://www.edureka.co/blog/rpa-developer-roles-and-responsibilities/.

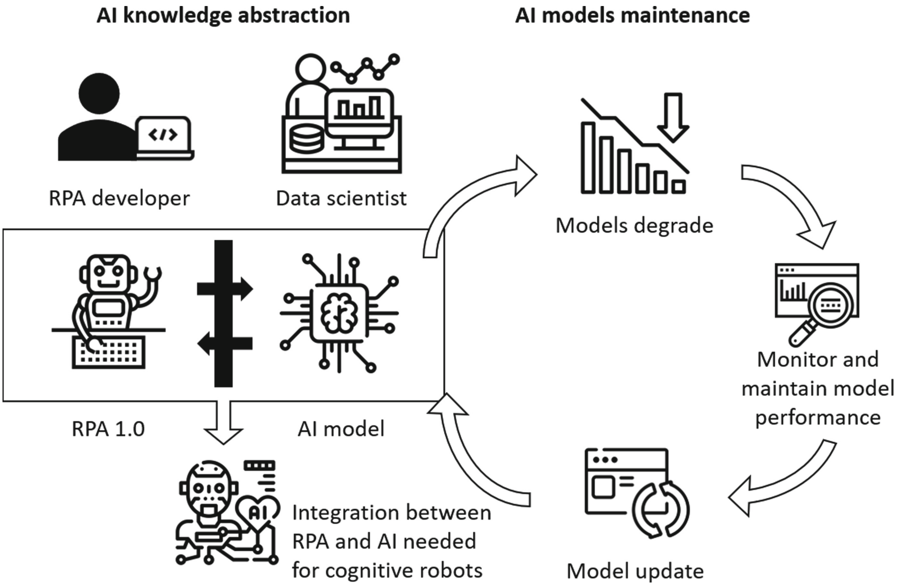

Fig. 1. Main challenges to be solved when making use of cognitive components.

briefly summarizes related work. Section 5 opens a discussion related to the project. Finally, Sect. 6 concludes the paper and describes future work.

2 Context

The RPA challenges described in Fig. 1 have also been pointed out by the industry, expressly, by Servinform S.A.[2] Servinform is a Spanish company dedicated to providing outsourcing services to other companies, mainly back-office processes automated with RPA. In the past years, they have identified the need to implement intelligent RPA solutions to empower business processes.

Integrating cognitive components in their processes allows automating tasks that previously required human intervention, i.e., aiming towards end-to-end process automation. For instance, the prediction of electricity consumption, considering that electricity use changes over time and consumption forecasts are of great value to utilities. For this purpose, a component is needed that determines what the consumption will be on the next bill, based on a customer's data history. Another example is related to document classification since companies typically use different formats when they issue documents. In this way, a component is required for classifying invoices or sales orders from different organizations with similar information but a different structure or style.

[2] https://www.servinform.es/.

The use of these components within an RPA process presents a series of challenges (cf. Fig. 1). As shown in the second challenge, a problem related to the AI components degradation and the reduction of ML model performance has been found. That is, this paper will focus its proposed architecture on the fact that the "*maintenance of a machine learning model involves regular updating to ensure that predictive effectiveness is not lost over time*" [6]. Therefore, Servinform, together with the IWT2 research group[3], tries to solve these challenges with the AIRPA research project, which will be described below.

3 Research Project

This section explains the AIRPA project. First, the initial objectives are presented. Second, the approach proposed in the project is detailed. Finally, the architecture to be developed and put into production is described.

3.1 Initial Goals

Based on Servinform industrial experience and the background within the RPA research line of the IWT2 group, the following goals are identified as the main ones of the AIRPA project:[4]

1. Create a collection of AI components to empower RPA solutions.
2. Create a nexus of union between both domains by understandably presenting the results in a platform for technical staff and business experts.
3. Automatize processes end-to-end that facilitate the integration between existing RPA solutions with AI components, reducing the need for human participation and decision-making.
4. Simplify and reduce the cost of access to RPA solutions powered with AI caused by licensing restrictions.
5. Enable RPA professionals who lack AI and ML skills to use AI components.
6. Define a lifecycle, development methodology, production, and integration roadmap of RPA solutions with AI components.
7. Verify the developed AIRPA framework in multiples realistic scenarios.
8. Integrate an AI components library in RPA solutions. Such integration seeks sustainability based on a cross-platform architecture orchestration independent of specific technologies and considers the degradation of AI over time.

3.2 Approach

The AIRPA project proposes a complete solution for the implementation of RPA processes using cognitive components, known in the industry as intelligent RPA processes.[5] To this end, it defines an architecture that supports its development

[3] https://www.iwt2.org/.

[4] As the project is under development, the realization of their goals are in progress.

[5] https://dlabs.ai/blog/rpa-2-0-how-to-achieve-the-highest-level-of-automation/.

and maintenance, divided into four modules (cf. Fig. 2): (1) *Document repository*, where components and robots are stored with their documentation and all their associated versions, (2) *Deployment manager*, which is used to control the deployment and version management of each component and the RPA robots, (3) *Tracking and exploitation panel*, which allows the visualization of the metrics and data associated with the execution of the processes, especially for the monitoring of the models that are associated with the cognitive components, and (4) *Control Room*, which allows for complete management of RPA processes, and handling cases in an execution state, KOs (i.e., failed situations), robots, equipment where they are executed, customized alerts, evidence capture, launches, or user roles, among others.

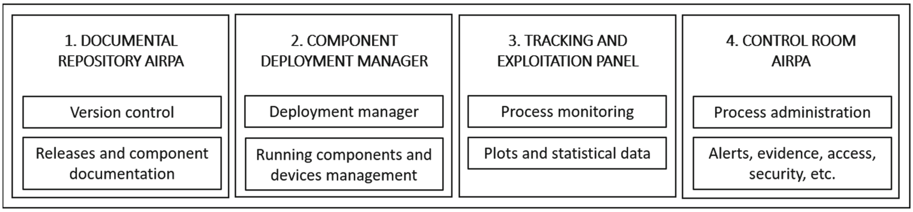

Fig. 2. Modules that compose the AIRPA platform.

The *control room* aims to provide comprehensive and centralized support to the intelligent RPA process management. This module resembles a customized state machine for each process, indicating which state a robot is in at any given moment and recording evidence of its transitions. The essential tasks of this module are (1) the collection of data needed for evidence capture and (2) the management and automated reporting of robots and tasks in which human intervention is required. The latter provides a differential value in intelligent RPA processes, thanks to labeling the data collected in daily work. In other words, the management of hybrid human-robot tasks allows the capture of data from the decisions which are made by humans, an essential task for the training of cognitive component models that will fully automate these tasks in the future. The collection of this data, together with the information reported by the cognitive components, feeds the *tracking and exploitation panel* for process reporting and obtaining valuable information of each process during its execution. It monitors, among other things, the performance of the models, facilitating their maintenance.

The first challenge focuses on the abstraction of the complexity of the development of an AI component by the RPA developer. For this purpose, *AIRPA components* are built, i.e., AI services with a microservices-based architecture that standardizes their service contract through a REST API. These components are designed as *wrappers* that allow the incorporation of ML models using the files previously exported in the data scientist's work environment. In this way, the RPA developer should only focus on consuming the methods offered by this API. Thus, any changes to the model will not affect its integration with the RPA solution.

Additionally, AI components designed as *wrappers* solve further problems. Generally, AI components in the context of intelligent RPA are implemented by commercial solutions (e.g., Amazon Web Services, Google, or Microsoft). These solutions pose several problems since a series of compatibility restrictions limit their use. They are not versatile enough to re-train the models from business data, and their customization becomes a rather complicated task. The *wrappers* component design allows the incorporation of proprietary Machine Learning (ML) models, solving these issues. This fact provides an added value since, although their use is not widespread in the context of RPA, open-source solutions are leading the main developments in the field of AI [12]. This design increases the specialization capacity of each model and reduces the cost of access caused by licensing limitations to AI components. Therefore, the *AIRPA components* of the AIRPA project, a library representing the first initial objective of the project, enable ML solutions such as classifications, anomaly detection, intelligent document processing, audio transcription, or sentiment analysis, among others.

3.3 Architecture

The AIRPA project proposes an architecture (cf. Fig. 3) for the execution of RPA processes that use cognitive components and that allows uploading, deploying, managing, and monitoring both robots and AI components. This architecture has different types of developments.

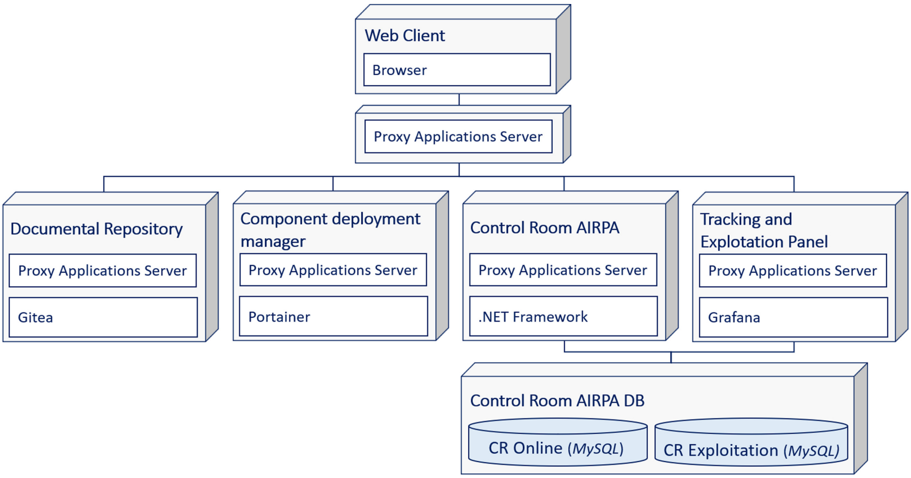

Fig. 3. Architecture diagram.

Firstly, the modules are based on several free software solutions as *Gitea* for the Document repository, *Portainer* for the Deployment manager, and *Grafana* for the Tracking and exploitation panel. Secondly, some solutions are based on customized development, as the *Control Room* implemented in *C#* with the *.NET* framework and the *MySQL* database. This module has two databases, the *CR Online* for the management of all information necessary to use *Control Room* (e.g., users, alerts, scheduled events, etc.) and the *CR Exploitation* with the data collection of all activity of the deployed robots and AI components to analyze them. Another customized solution is the web client developed with *ASP.NET Web Pages (Razor)* and offers access to all the modules that comprise the AIRPA architecture from a unique site that permits login to the whole system. Finally, the creation of *wrappers* for the *AIRPA Components* is a custom design implemented in *Python* language with *Django REST Framework*. In its construction, several specific libraries from the field of data processing and ML are used to facilitate feature engineering and the incorporation of ML models from *Scikit Learn*, *TensorFlow*, and *PyTorch* based on a service contract.

3.4 Achievements of Goals

The current status of the AIRPA platform shows the degree of accomplishment of the initial objectives of the project.

4 Related Work

In the current industry, some platforms aim to solve problems related to the one addressed in this paper. However, they are oriented to different perspectives.

Google Cloud[6], Amazon[7] and Azure[8] allow deploying and monitoring pre-created or custom cognitive components. These platforms can monitor and detect cognitive degradation. However, they are not unified to be used in RPA. It should be noted that they offer RPA integration but deploying and monitoring AI components is in an isolation system separated from the monitoring of RPA.

That makes the monitoring more complex due to the use of AI models, e.g., Blue Prism[9] allows the use of cognitive components deployed in Google Cloud, where they are monitored. Nonetheless, it forces the use of both (1) the Google platform to monitor cognitive components and (2) the Blue Prism platform to monitoring the non-cognitive ones. Similarly, UIPath[10] platform offers a service called *AI Center*. The service allows deploying cognitive components and monitoring them but, unlike the AIRPA platform, it does not detect the degradation of components. Moreover, *AI Center* is a proprietary solution and only accessible for use in the UIPath technology stack (i.e., UIPath Studio, AI Center, and

[6] https://cloud.google.com/vertex-ai.
[7] https://aws.amazon.com/sagemaker/.
[8] https://azure.microsoft.com/en-gb/services/machine-learning/.
[9] https://www.blueprism.com/.
[10] https://www.uipath.com/.

Table 1. Table of the completed initials goals.

Goal	State	Comments
1	Completed	An AI components library described in Sect. 3.2 is developed
2	Completed	Thanks to the *tracking and exploitation panel*, graphs and statistics can be presented that are easy to understand for business experts
3	Completed	The use of the *control room* gives full support to process automation and, therefore, to end-to-end automation, which reduces the number of people needed to solve process cases and reduces costs
4	Completed	The proposed design enables to use open-source solutions, which actually lead the field of AI, increasing the specialization capacity of each model and reducing the cost of access caused by licensing limitation
5	Completed	The proposed architecture abstracts RPA developer from AI models development and maintenance, through the use of *wrappers*
6	In validation	A robotization guide is being defined to be applied for the design, development and maintenance of each component, still pending to be validated
7	In progress	The AIRPA project is still in the validation phase, but has been tested with some real scenarios as shown in Sect. 5
8	Completed	AIRPA architecture allows the integration of AI components in RPA processes from controlling the status of the robots to monitoring their performance.

Orchestrator). In other words, the existing RPA platforms do not support the detection of model degradation in AI components. So it is necessary to navigate to AI service platforms providers to consult this information. Moreover, since AI services are not integrated with RPA platforms, it is necessary to manage logins, tokens, etc. between the two platforms. Furthermore, AIRPA differentiates itself from other platforms by the use of AI components as *wrappers*. With these, APIs through AI components are offered, they follow the same specification and, therefore, RPA developers only need to know this specification to implement the AI-RPA integration (Table 1).

Besides these commercial solutions, there are scientific research proposals that study the RPA and the AI field [4,13]. Some of them use AI not to build components, but for early stages in the RPA lifecycle, in the process discovery phases [3,7,10,11]. Nevertheless, others deal with the application of AI on RPA processes, such as [1], which studies key open research challenges that exist in the combination of RPA with AI, or [9], which proposes a dynamic taxonomy for intelligent RPA components. There are other proposals to improve the RPA architecture. For example, [8] elaborates on optimizing the deployment architecture of the RPA components. Nevertheless, to the best of our knowledge, there are no proposals that focus on the operation and maintenance phase of the robots, such as AIRPA. At a glance, unlike existing proposals, AIRPA proposes a system to manage and intensively monitor cognitive components, separating the AI development from the RPA developer. All these are carried out easily and integrally in one platform.

5 Results

The AIRPA project is still in the validation phase, and therefore the final results may vary after it. The platform has been validated in several real scenarios that require the incorporation of cognitive actions. In this way, the aim is to evaluate the AIRPA operation applied to different business areas, such as energy or telecommunications. One of these validation has been carried out specifically on the consumption prediction use case shown in Sect. 2. In this case, the generation of automatic predictions from historical data, without the need for human interaction, represents a considerable improvement. It is performed within Servinform's operation area by taking measurements during one month of the Average Handle Time (AHT) before and after implementing the AIRPA platform. Initially, an AHT of 9 min was obtained, which was improved after implementing AIRPA by 75%, resulting in a final AHT of 2 min and 15 s. Even though the results are preliminary, the platform significantly increased control over the process, which suggests promising results. AIRPA is planned to be a platform and a methodological strategy followed by Servinform S.A. and its entire Consulting and Innovation area.

The conduction of the AIRPA project has lead to some lessons learned. After defining and implementing AIRPA architecture, we have realized that it is very focused on high-demand situations in terms of deployments. In some real scenarios, the level of demand for deployments is lower because they are less frequent. Therefore, in that cases, such a complex architecture is not necessary, and a deployment pipeline would be sufficient. In addition, the experience with different real cases showed limited use of the shared collection of RPA components. This situation was mainly since each case required a level of customization that neglects its transversality. Nonetheless, the *wrapper* design of the components enables easy customization from one business case to another.

6 Conclusions and Future Work

This paper presented the AIRPA project that aims to improve RPA thanks to the integration of AI, expanding the automation of the end-to-end processes. The project allows (1) the abstraction of the data scientist from the RPA developer and (2) extensive monitoring of robots and AI components to detect the degradation of the cognitive components. To solve these challenges, AIRPA has been built with a microservice architecture and the standardization of a service contract thanks to an API REST. These components are designed as *wrappers* to facilitate their integration. This architecture is composed of 4 modules: the document repository, the deployment manager, the AIRPA Control Room, and the tracking and exploitation panel. All these modules allow the RPA developers and the data expert to work independently. In addition, thanks to the Control Room and the tracking panel, the detection of AI degradation is possible.

Although the project is still in progress, it offers promising preliminary results and possible lines of future research work. (1) The project, being research-based, could have performance improvements and better functionalities. (2) The component library can be extended by adding new functionalities. (3) Currently, the data scientist requires an existing *wrapper* component before loading a model. The need to avoid the dependency of the data scientist from the RPA developer role and ease the modification of components is identified, so the use of preloaded *wrappers* is proposed as future work. (4) The current *Control Room* is a custom state machine (cf. Sect. 3.2) and the robot is the one in charge of modifying the state. As future work, we plan to extend the component behavior to automatically generate multiple state changes and evidence transitions records.

References

1. Chakraborti, T., Isahagian, V., Khalaf, R., Khazaeni, Y., Muthusamy, V., Rizk, Y., Unuvar, M.: From robotic process automation to intelligent process automation: emerging trends. In: Internation Conference on Business Process Management 2020 RPA Forum. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58779-6
2. Jha, N., Prashar, D., Nagpal, A.: Combining artificial intelligence with robotic process automation—an intelligent automation approach. In: Ahmed, K.R., Hassanien, A.E. (eds.) Deep Learning and Big Data for Intelligent Transportation. SCI, vol. 945, pp. 245–264. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-65661-4_12
3. Jimenez-Ramirez, A., Reijers, H.A., Barba, I., Del Valle, C.: A method to improve the early stages of the robotic process automation lifecycle. In: Giorgini, P., Weber, B. (eds.) CAiSE 2019. LNCS, vol. 11483, pp. 446–461. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-21290-2_28
4. Hollebeek, L.D., Sprott, D.E., Brady,M.K.: Rise of the machines? Customer engagement in automated service interactions. J. Serv. Res. **24**(1), 3–8 (2021). https://doi.org/10.1177/1094670520975110, https://www.sciencedirect.com/science/article/pii/0166361596000139
5. Le Clair, C., O'Donnell, G., Lipson, A., Lynch, D.: The Forrester Wave™: Robotic Process Automation, Q4 2019. The Forrester Wave (2019)
6. Leevy, J.L., Khoshgoftaar, T.M., Bauder, R.A., Seliya, N.: The effect of time on the maintenance of a predictive model. In: 2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA), pp. 1891–1896. IEEE (2019)
7. Leno, V., Polyvyanyy, A., Dumas, M., La Rosa, M., Maggi, F.M.: Robotic process mining vision and challenges. Bus. Inf. Syst. Eng. **63**,301–314 (2020)
8. Mahala, G., Sindhgatta, R., Dam, H.K., Ghose, A.: Designing optimal robotic process automation architectures. In: Kafeza, E., Benatallah, B., Martinelli, F., Hacid, H., Bouguettaya, A., Motahari, H. (eds.) ICSOC 2020. LNCS, vol. 12571, pp. 448–456. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-65310-1_32

9. Martínez-Rojas, A., Barba, I., Enríquez, J.G.: Towards a taxonomy of cognitive RPA components. In: Asatiani, A., García, J.M., Helander, N., Jiménez-Ramírez, A., Koschmider, A., Mendling, J., Meroni, G., Reijers, H.A. (eds.) BPM 2020. LNBIP, vol. 393, pp. 161–175. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58779-6_11
10. Martins, P., Sá, F., Morgado, F., Cunha, C.: Using machine learning for cognitive Robotic Process Automation (RPA). In: Iberian Conference on Information Systems and Technologies (CISTI) (2020)
11. Singh, M.K., Raghavendra, D., Pandian, D., Sadana, A.: Surface automation - interacting with applications using Black box approach. In: International Conference for Convergence in Technology (I2CT) (2021)
12. Sonnenburg, S., et al.: The need for open source software in machine learning. J. Mach. Learn. Res. **8**,2443–2466 (2007)
13. Willcocks, L., Lacity, M., Craig, A.: Robotic process automation: strategic transformation lever for global business services? J. Inf. Technol. Teach. Cases **7**(1), 17–28 (2017)