IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 27 April 2023, accepted 8 May 2023, date of publication 12 May 2023, date of current version 17 May 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3275932

RESEARCH ARTICLE

ARivaT: A Tool for Automated Generation of Riva-Based Business Process Architecture Diagrams

OSAMA AL-HAJ HASSAN[1], AHMAD SAMHAN[2], SUHAIR ALHAJHASSAN[3], AND RAWAD HAMMAD[4]

[1]Computer Science Department, Princess Sumaya University for Technology, Amman 11941, Jordan
[2]Software Engineering Department, Zarqa University, Zarqa 13110, Jordan
[3]Software Engineering Department, Applied Science Private University, Amman 11931, Jordan
[4]Computer Science and Digital Technologies Department, University of East London, E16 2RD London, U.K.

Corresponding author: Ahmad Samhan (ahmad.samhan@zu.edu.jo)

ABSTRACT The Riva approach is used to develop an object-based Business Process Architecture (BPA) that helps in capturing the full organizational strategies. However, due to the lack of effective tools that can generate Riva BPA models from available artifacts, producing such models is time consuming non-automatic process and it requires an exhaustive manual validation to ensure the development of error-free models. This paper introduces a novel and domain-independent tool titled 'ARivaT' to automatically generate Riva BPA models from available knowledge assets such as Units of Work diagrams. ARivaT is underpinned by a step-by-step methodological approach that automates the process of generating Units of Work, First-Cut, and Second-Cut Process Architecture Diagrams. It also provides stakeholders with insightful explanations to allow a precise understanding of business processes and workflow. Furthermore, ARivaT employs a rule-based mechanism to seamlessly validate all the generated Riva BPA models. A case study-based approach has been followed to evaluate the applicability of ARivaT to derive Riva BPA diagrams. The results of our experiment are promising since 82 percent time saving has been recorded when using ARivaT to generate Riva BPA diagrams in comparison to using general drawing tools such as MS Word and MS Visio. In addition, 69 percent time saving has been noted when performing the same task with ARivaT as opposed to specialized tools such as Camunda.

INDEX TERMS Riva, business process architecture, business process management, software requirements, automation, forward engineering.

I. INTRODUCTION

A business process is defined as a set of related activities designed to achieve the predetermined goal(s), which entails delivering well-specified value to stakeholders [1]. The recent move towards a process-centric culture required a process perspective method that can identify organization's business flow at process level [2]. This process perspective leads to a well-defined specifications, which brings more clarity to business owners [3]. Consequently, various business process platforms and tools for business process modelling, management and orchestration have been developed. Riva [4] is one of the BPA modeling methods that can be realized through these tools. Hence, we plan to investigate current tools used to draw BPA diagrams for the purpose of answering the following research questions, **RQ1:** Are there existing tools specifically tailored for Riva-based modeling specifications? **RQ2:** If not, would developing a Riva-specific tool result in improvements in creating BPA diagrams?

The associate editor coordinating the review of this manuscript and approving it for publication was Davide Aloini.

A. LITERATURE REVIEW

Our research is related to the areas of Business Process Management (BMP), Business Process Architecture (BPA) and Riva BPA Modelling. Therefore, it is valuable to briefly mention a literature review of these areas. Business Process Management (BPM) focuses on improving, managing, and controlling business processes to help an organization achieve its goals [5]. These goals can span operational, management, or strategic levels [6]. Fulfilling these goals can be done by the elicitation, definition, and modeling of the key related business processes which in turn improves organization performance [7]. Depicting a business process not only improves performance but also boosts productivity, supports agility, and reduces expenses to complete a task [8]. Representing business processes can be attained by employing Business Process Architecture (BPA) which is considered among the essential viewpoints used to enable a common view of an organization's processes and their interrelationships [9]. BPA can be utilized to group related tasks into essential business processes that can then be modelled [10]. The work in [11] and [12] examined the available strategies and guidelines for constructing a BPA, identifying the most effective strategies and how they are applied.

Object-based BPA modeling is among the popular BPA modeling approaches and one key representative of this category is Riva [4] which has received widespread recognition. Through Riva, the BPA is developed by examining business entities (objects) and their dynamic relationships. This helps in producing three main Process Architecture Diagrams (PADs) that are employed in describing business processes, namely, Units of Work (UOWs), First-Cut Process Architecture Diagram (First-Cut PAD), and Second-Cut Process Architecture Diagram (Second-Cut PAD).

Riva has been used in various areas, such as deriving software services for Service Oriented Architecture (SOA) [13], semantic-based derivation of enterprise information architecture [14], business process change management [15], system of systems change management [16], and cancer care informatics [17], [18].

B. RESEARCH GAP

Once the BPA is established, the technological solutions needed to support it become clear [19]. Example of these technological solutions is the various modeling tools that can be exploited to draw Riva BPA diagrams. For example, Ould has proposed a Visio 2003 stencil called the Riva Visio Stencil [20]. It provides Riva customized notations that can be used to model Riva-driven BPA models. However, this remains a manual process and this stencil is outdated and it is not available at the time of conducting this research. Tools for drawing BPA diagrams can be either general purpose drawing tools such as MS Word and MS Visio or more specialized tools such as Camunda [21]. These tools have their own strength points and shortcomings.

General-purpose tools have common shapes which the user can employ in drawing Riva BPA diagrams. The general-purpose tools are beneficial in various situations. When the complexity of business processes is low, then resulting Riva BPA diagrams are expected to be less complex and therefore using general purpose drawing tools can be sufficient. Also, when the size of organization is small, then the complexity of their business processes tends to be low and therefore general purpose tools can be utilized in this case. In addition, organizations with low budget cannot afford buying sophisticated tools for managing their business processes and accordingly they can rely on general-purpose drawing tools. Having the previous cases in mind, general-purpose tools fall short in cases where size of the organization is large and its business processes are complex.

Specialized tools have Riva BPA notations available for the user to use. In addition specialized tools have drawing features that facilitate drawing Riva BPA diagrams such as available grid and snapping to grid points. However, in the cases where specialized tools cost money, low-budget organization might not be able to afford them. Furthermore, none of these tools is specifically tailored to Riva which results in having the user manually drawing First-Cut PAD and Second-Cut PAD. This is problematic for five reasons. First, theoretical rules can be applied to generate First-Cut PAD and Second-Cut PAD from their corresponding UOWs diagram, and therefore, it is a cumbersome and time consuming task for the user to manually draw them. Second, the rules that bound First-Cut PAD and Second-Cut PAD to their corresponding UOWs diagram naturally force First-Cut PAD and Second-Cut PAD to be almost twice the size of their corresponding UOWs diagram. Consequently, user mission to draw First-Cut PAD and Second-Cut PAD becomes more complicated and time intensive. Third, if the organization's business processes are complex, then there corresponding Riva BPA diagrams might be large and complex as well and this in turn makes manual drawing of First-Cut PAD and Second-Cut PAD even more challenging and eats up more time. Fourth, the produced Riva-based BPA models are expected to have various numbers of errors due to the demanding nature of extracting information from organizational assets as well as the human errors that might happen during following that extraction process. Fifth, tools that are not specifically customized for Riva lack analysis capabilities that allow system modelers to focus on processes that are vital in the organization's workflow. To elaborate on that, problems might arise in the way team members and system modelers will interpret the organizational information. For instance, various analytical activities are required to understand: (1) the importance of each organizational component and (2) the most resource-consuming ones while Riva-based models are being developed. Such analysis should also be linked with the wider context of applying the business process architecture diagram and its constituent steps. For instance, deriving the First-Cut PAD from its corresponding UOWs diagram requires a thorough insight from a business perspective as well as BPA perspective. The existence of precise rules to develop the BPA diagram is not always sufficient to produce current BPA diagrams because this is frequently mixed with the unclear understanding of organizational processes. According to [22], various system analysts/modelers mix what they have in the business, i.e., the as-is process, with what they want to have, i.e., the to-be process. This confusion is caused by different reasons including the misunderstanding of business needs and procedures and the need to develop more logical processes for daily business activities. This implies that analysis of BPA diagrams are needed to support decision makers in enhancing their business workflow.

Hence, we realized that there is a room for improvement via forward engineering the process of developing BPA models. In addition, budget wise, ARivaT helps both organizations with low and high budgets because it is provided for free use and offered as an open source software [23] which helps in continuous development by research community. Finally, the substantial move towards adopting a business process-oriented culture in organizations led to significantly rich and complicated business processes and consequently complicated Riva-based diagrams. Thus, automating the process of developing Riva Business Process Architecture models is essential for successful and competitive organizations. The current tools assist user in drawing BPA models through providing required annotations and grid features. However, to the best of our knowledge, they do not offer forward engineering features such as auto generation of First-Cut PAD and Second-Cut PAD from their corresponding UOWs diagram.

C. RESEARCH OBJECTIVES

Based on the above-identified problem, we emphasized the eminent need for developing a tool that facilitates and automates the process of generating Riva business process architecture diagrams. In this paper, we propose a tool (**A**utomatic **Riva T**ool) (ARivaT) for automated drawing and generation of Riva diagrams. The objectives for this research are detailed as follows:

1) Aiming to save considerable time and effort when drawing Riva-based diagrams by using forward engineering concepts. The target is to simplify the key concerns of developing the tool and its constituent components from a high-level business view.
2) Envisioning a tool specifically tailored to generating Riva diagrams which can help in setting the standards of drawing such diagrams and their shapes.
3) Looking to handle the details of the generated BPA diagrams, hence the time of users can be directed to producing a high-quality diagram rather than focusing on details.
4) Planning to generate machine-readable code for BPA diagrams which provides integration with other systems if needed.
5) Finally, the tool is foreseen to provide various charts to help system modelers to further analyze Riva BPA diagrams and extract insights from them.

The rest of the manuscript is organized as follows. Section II is the methods section which discusses ARivaT design, architecture, development, and validation. It is followed by the results section III in which ARivaT is evaluated by conducting an experiment and explaining how ARivaT is able to extract several insights from Riva diagrams. Following that is the discussion section IV which highlights several findings, outlines points of strength, limitations, and future direction in ARivaT. Finally, section V concludes the paper.

II. METHODS

The research method we follow during our work is the design science research methodology. We stated the problem identification and research objectives in Section I. In this section, we will cover the design and development stage of ARivaT. Then, we explain the demonstration stage by using ARivaT to represent the higher education use case. After that, we go through the evaluation stage by evaluating ARivaT using quantitative data.

To respond to the early-identified research problem, further studies have been conducted on related literature, especially deriving BPA models for different organizations including [24], [25] among others. The common approach used across all investigated papers is using a step-by-step process to deal with one concern at a time (e.g., UOWs, First-Cut PAD, etc.) to reach the final goal, i.e., modelling the BPA diagrams. Based on the earlier analysis, we decided to adopt a component-based architecture to develop ARivaT due to the following reasons. First, it allows decomposing the system into a set of logical or functional constituent modules where each one focuses on a certain number of features. Second, the ability to have well-defined and formal communication interfaces between the constituent parts of the tool, will improve reusability, maintainability, to mention but a few [26]. Consequently, the authors decided to pay more attention to the following three types of Riva business process architecture diagrams, namely, UOWs diagram, First-Cut PAD, and Second-Cut PAD.

These three types of diagrams are key to Riva BPA modelling. In general, the Riva-based BPA modeling approach is performed by following these steps (1) determining organizational boundaries and business, (2) identifying Essential Business Entities (EBEs), which are business-related entities. (3) Identifying Units of Work (UOWs) by applying filters to the EBEs list to determine the processes needed to manage their lifetimes. (4) Modelling the UOWs diagram by recognizing the dynamic relationships between the identified UOWs. (5) Modelling the First-Cut PAD, here, Case Process (CP), Case Management Process (CMP), and Case Strategy Process (CSP) are assumed and modeled, along with their interrelationship. (6) Modelling the Second-Cut PAD by applying Ould's heuristics on the First-Cut PAD to produce a more compact and realistic Second-Cut PAD. Figure 1 shows the previous steps. Also, generating these three Riva diagrams is among the major tasks in ARivaT and this can be seen in Figure 2 which reveals the main use case diagram for ARivaT and shows its key functionalities. According to the figure, users can generate Riva diagrams, namely, UOWs diagram, First-Cut PAD and Second-Cut PAD. In addition, ARivaT integration and coexistence with other technologies remain possible due to the ability to generate machine-readable models, i.e., XML-based specifications.

A. ARivaT ARCHITECTURE

The process of generating the previously mentioned BPA diagrams within ARivaT is illustrated in Figure 3. The figure depicts the UML component-based architecture of ARivaT as follows: (1) diagram loader component, (2) layout editor component, (3) analysis engine component, (4) layout engine component, (5) utility component and (6) mapping services component. The layout editor component allows users to edit the previously listed three types of Riva diagrams. It uses the interface provided by the diagram loader component which facilitates the process of saving diagrams as XML files and loading them back to the layout editor component whenever needed. The layout editor component utilizes the interface provided by the layout engine component which enables users to either: (a) use drag/drop capabilities to design diagrams, or (b) to rely on the automatic layout built in ARivaT. Moreover, the layout engine component consumes the services offered by the Mapping Services component, which is a core component of our tool. The mapping services component consists of the following three subcomponents. First, the UOWs diagram to First-Cut PAD mapper subcomponent which facilitates the translation from UOWs diagram to its corresponding First-Cut PAD. Second, the First-Cut PAD to Second-Cut PAD mapper subcomponent, which governs the process of generating Second-Cut PAD from its corresponding First-Cut PAD. This subcomponent has a dependency relationship with the UOWs Diagram to First-Cut PAD mapper, as the resulting First-Cut PAD of this subcomponent is used in the First-Cut PAD to Second-Cut PAD mapper subcomponent. The First-Cut PAD to Second-Cut PAD mapper subcomponent also has a dependency relationship with Ould's Heuristics checker subcomponent, which guarantees a sound process of mapping First-Cut PAD to Second-Cut PAD based on Ould's heuristics. Finally, the layout editor component employs the services provided by the analysis engine component to enable users to have useful insights from Riva diagrams, which is useful for stakeholders to understand their business flow. To enhance the usability of ARivaT, a few diagram-manipulation supportive features including printing diagrams, zooming in and out, undo/redo, etc. are provided by the utility component, which can be accessed seamlessly from the user interface.

B. ARivaT DEVELOPMENT

This section provides an in-depth view of ARivaT development throughout its various stages. However, due to the intense complexity of the tool, this section will focus on the algorithms as well as the development of its key features. As described earlier, ARivaT aims at generating Riva-based business process architecture diagrams. There are various concerns around developing this tool, nonetheless, the key one is providing a seamless transition between the following three stages, i.e., UOWs diagram, First-Cut PAD and Second-Cut PAD. To achieve this goal, the researchers utilized mxGraph library [27] which has several APIs that can work with various programming languages to meet users' requirements. The main reason behind using mxGraph library is to liaise between the user interface, in a very friendly way, and the different components, especially the layout editor component. The selection of mxGraph is very well aligned with the early-described ARivaT design specifications (i.e., web-based tool) as mxGraph's JavaScript APIs [28] have been used to establish the connection between the web interface and the backend of the tool itself. The key algorithms along with ARivaT key features will be briefly described in the next sub-sections.

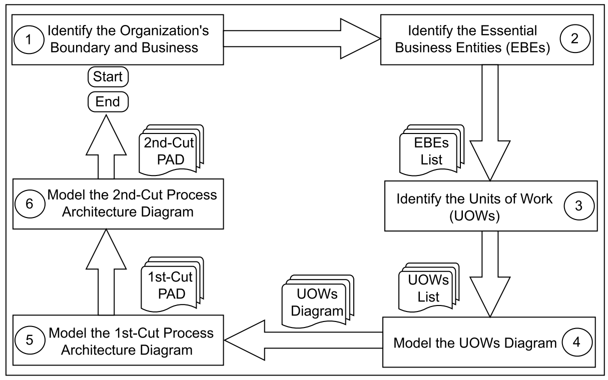


FIGURE 1. Main steps of the Riva-based BPA approach.

1) ARivaT GENERAL FEATURES

Following the use case diagram and its key features explained in the beginning of this section, ARivaT provides users with the capability of creating a new diagram starting from the UOWs diagram. The connection between the layout editor component and the diagram loader component, as described in Sub-section II-A, allows users to save existing diagrams in mxGraph XML format and import them later on, which is a very useful feature to be used throughout the modelling exercise. Once users create the Units of Work diagram, ARivaT should allow them to navigate between the early-identified three diagrams, i.e., UOWs, First-Cut PAD and Second-Cut PAD. Moreover, users should be able to visualize the characteristics of the diagram to understand it and also to adjust its edges to improve the organization of the produced models/diagrams. Various serverless technologies/frameworks have been used to develop drag-and-drop options to minimize the time used to develop Riva-based models.

The above-developed UOWs diagram should be automatically transformed into First-Cut PAD and Second-Cut PAD seamlessly. The researchers utilized the early-described drawing edges feature to improve the diagram automatically. To do so, each edge in the UOWs diagram is recognized in the following three capacities: (1) the name of the interaction between the source and destination Units of Work, (2) the

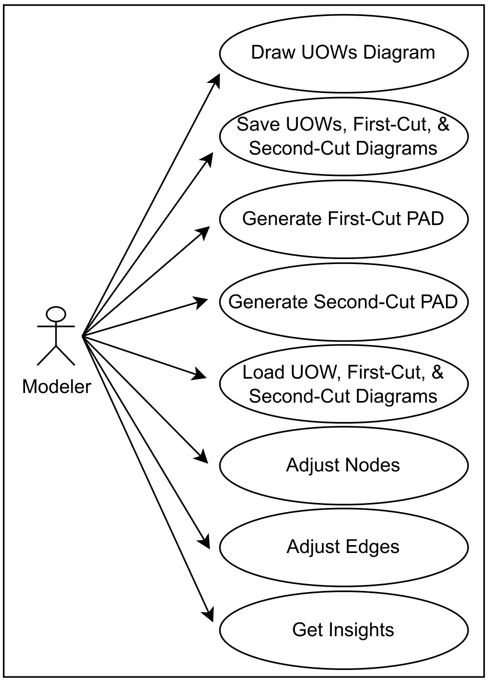


FIGURE 2. ARivaT use case diagram.

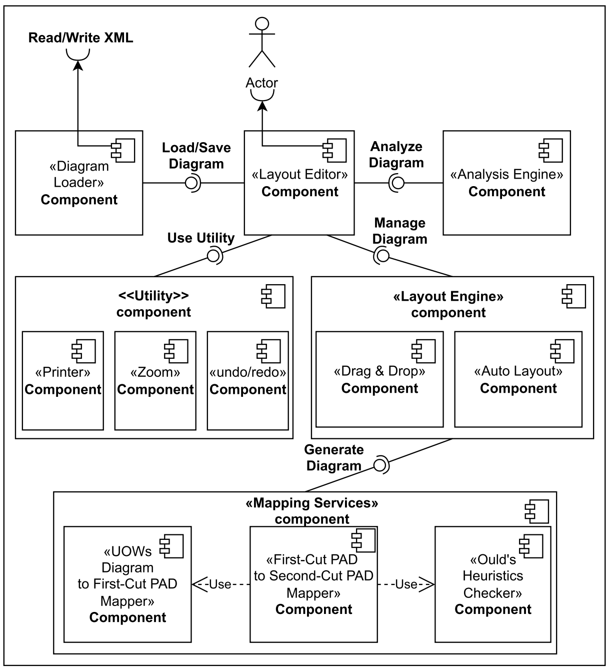


FIGURE 3. ARivaT architecture.

type of this interaction, i.e., Task Force (TF) or Service (S), and (3) the multiplicity of the interaction. Furthermore, users can change the properties of nodes and edges in the UOWs diagram. Such properties include, for instance, the ability to control the UOW label and whether a Case Strategy Process (CSP) should be included for this particular UOW when the First-Cut PAD is generated. Other properties that can be controlled by the user include the relationship label, the relationship name, the relationship type (Task Force or Service) and the relationship multiplicity. The above properties are essential for a good usability level according to users' feedback. However, the following two features are intrinsic to ARivaT. First, the user can control whether when generating First-Cut PAD, a negotiate relationship between the target CMP and the source CMP nodes should be included or not. Second, the user can determine whether a negotiate relationship should be included between the target CMP and the source CP when the First-Cut PAD is generated. Additionally, users can change edge type to 'Straight', 'Orthogonal', or 'Curved'. It is worth mentioning that one of the powerful features of ARivaT is that it translates Riva diagrams to machine-readable XML code which can be saved/loaded per user desire. This machine-readable code is useful for the research community as it can be used as the bases for research work related to Riva BPA diagrams. Figure 4 shows XML representation for part of a general First-Cut PAD (5).

2) ADJUST EDGES FEATURE

To ensure a smooth development for UOWs diagram, First-Cut PAD and Second-Cut PAD, ARivaT possesses a handy feature called Adjust Edges. If the user turned on this feature, nodes' movement will rearrange edges connectivity to simplify the connection between source and destination and ensure the shortest straight-line distance between them. It is recommended to turn this feature off to allow the user to perform manual little adjustments, such as reconnecting edges to different connection points (constraints) without having the edges readjust automatically. The Adjust Edge feature algorithm is explained in algorithm 1, and it is further described as follows.

If a node is moved while the 'Adjust Edges' feature is turned on, then when the node settles in its new place, calculations are computed for each ingoing edge of the node. For each edge connected to the node the user tries to move, there are 'n' constraint points around the moved node such that the minimum value of 'n' is 4 (1 constraint point per one side of node). So, the Euclidean distance between (x_1, y_1) position of each of the constraints surrounding the moved node and (x_2, y_2) position of the source of an ingoing edge is computed (lines 7 and 8). In lines 9 and 10, all the previous distances are stored in an array and then sorted in ascending order. Afterwards, in line 11, the ingoing edge will be reconnected to the constraint point which yielded the minimum distance. This constraint point is marked 'occupied' in line 12. Similarly, the same process is repeated for outgoing edges of the moved node, except that (x_2, y_2) will refer to the position of the target of each outgoing edge. The UOWs diagram generated by ARivaT, while the adjust edges feature is turned on, for the higher education process architecture is shown in Figure 6.

3) MAPPING UNITS OF WORK DIAGRAM TO FIRST-CUT DIAGRAM

In this particular part of the tool, the UOWs diagram is converted to a First-Cut PAD. To do so, the UOWs diagram

```xml
<mxGraphModel>
    <root>
        <mxCell id="0"/>
        <mxCell id="1" parent="0"/>
        <mxCell id="v12_cp" value="CP: Handle UOW1" vertex="1" parent="1">
            <mxGeometry x="201" y="1" width="84" height="84" as="geometry"/>
        </mxCell>
        <mxCell id="v12_cmp" value="CMP: Manage the flow of UOW1 cases" vertex="1" parent="1">
            <mxGeometry x="1" y="1" width="84" height="84" as="geometry"/>
        </mxCell>
        <mxCell id="e_v12_cmp_v12_cp_start" value="start" edge="1" parent="1" source="v12_cmp" target="v12_cp">
            <mxGeometry width="19" height="20" relative="1" as="geometry"/>
        </mxCell>
        <mxCell id="e_v12_cmp_v12_cp_start_child" value="A"
            style="rounded=1;shape=ellipse;" vertex="1" parent="e_v12_cmp_v12_cp_start">
            <mxGeometry x="-1" width="20" height="20" relative="1" as="geometry">
                <mxPoint y="-10" as="offset"/>
            </mxGeometry>
        </mxCell>
    </root>
</mxGraphModel>
```

FIGURE 4. **mxGraph XML representation of diagram in Figure 5.**

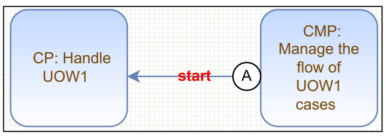


FIGURE 5. **Part of a general first-cut PAD. Its XML representation is shown in Figure 4.**

is taken as an input for the whole process. Then the following rules are followed to generate the First-Cut PAD accordingly:

1) Each Unit of Work is mapped to a CP node and a CMP Node.
2) Each Unit of Work might result in a CSP node based on user choice.
3) Each interaction between two Units of Work is mapped to the following mandatory interactions: (1) A 'start' interaction initiated from each CMP node to its corresponding CP node, (2) A 'request' interaction connecting the requesting CP node to the requested CMP node, (3) A 'deliver' interaction starting from the requested CP node and going to the requesting CP node.
4) Based on the user's choice, an optional 'negotiate' interaction can start from the requested CMP node to the requesting CMP node.
5) User can choose to have a 'negotiate' interaction drawn from the requested CMP node to the requesting CP node.
6) If a CSP node exists, then two interactions are generated, (1) a 'direct' interaction starts from CSP node and reaches its corresponding CMP node, and (2) a 'guide' interaction begins at CSP node and connects to its corresponding CP node.

Once the user clicks on the 'Generate First-Cut Diagram' in the 'Generate Diagram' drop-down list, ARivaT will generate the First-Cut PAD from the UOWs diagram. As we

Algorithm 1 Adjust Edges Feature
Input: Node: moved node, graph: model of diagram, constPoints: An array of constraint points around node that is moved
Output: Edges are reconnected to appropriate constraint points

let outEdges = A List of outgoing edges of moved node
let inEdges = A List of ingoing edges of moved node
let constTaken = A List to track if a given constraint point is occupied with an edge.
for (each Edge E_i in inEdges) **do**
    let x1 = E_i.source.xCoordinate;
    let y1 = E_i.source.yCoordinate;
    let distance = new Array(constPoints.length);
    euclideanDist(constPoints, x1, y1, distance);
    let constPointsPos = new Array(distance.length);
    asscendOrder(constPointsPos, distance);
    graph.connectCell(E_i, node, $constPoints_z$);
    constTaken[$constPoints_z$] = 1;
end for
repeat the above code for outgoing edges such that targets of outgoing edges are considered instead of their sources

show later on, the initial Second-Cut PAD is identical to the First-Cut PAD. So, for conserving space reasons, we show only one of them (the initial Second-Cut PAD) in Figure 7. So, Figure 7 resembles the First-Cut PAD which corresponds to the UOWs illustrated in Figure 6 diagram of the higher education process architecture. In this figure, you can see all the generated nodes, the relationships between them as well as the labels in addition to other details. To complete the process of generating Riva-based business process architecture model, the initial Second-Cut PAD must be generated from First-Cut PAD as it will be shown in Section II-B4.

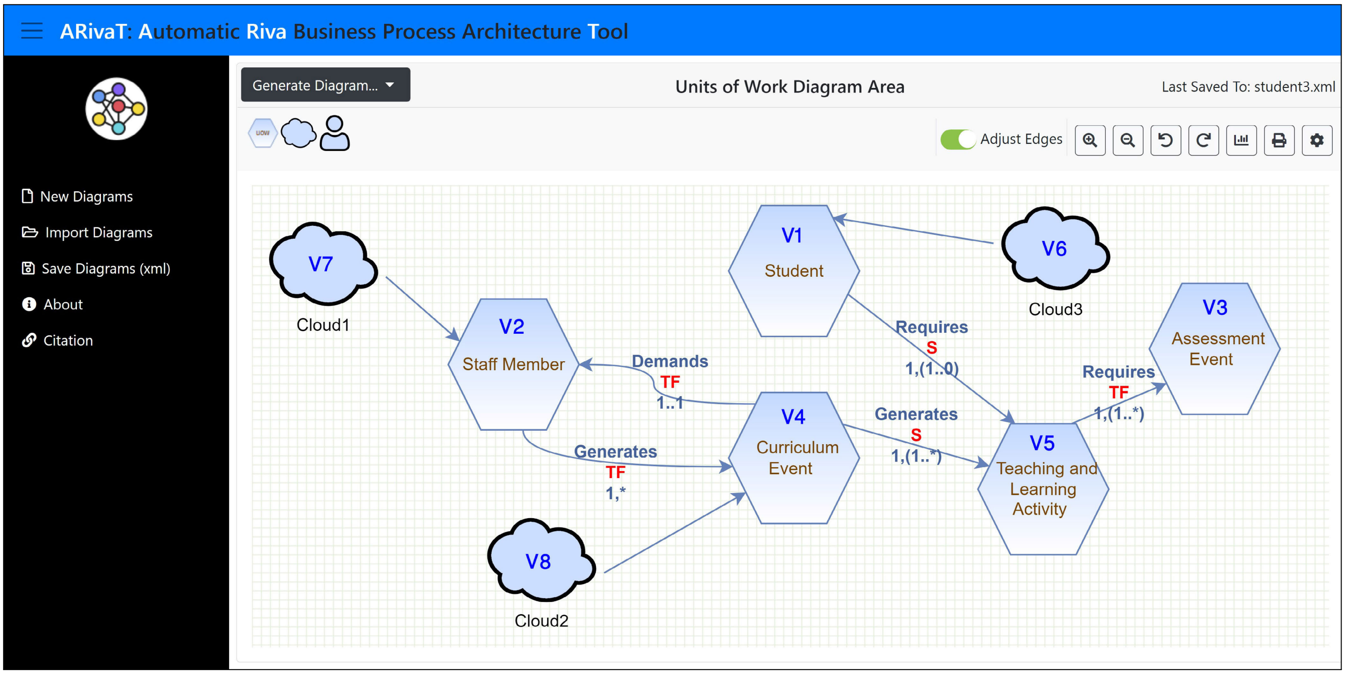


FIGURE 6. UOWs diagram generated using ARivaT for the higher education process architecture.

4) APPLYING OULD's HEURISTICS

In the majority of cases, the user needs to implement adjustments to the First-Cut PAD to make it closely reflect reality. Applying these adjustments (i.e., Ould's Heuristics) results in the Second-Cut PAD. Some of these changes can only be realized manually based on the user's decision, while others can be performed semi-automatically with the assistance of ARivaT. The first heuristic is folding a task force CMP into its requesting CP. This heuristic applies to CMPs that receive a single task force interaction. Then the user can right-click on the CMP to be folded and choose 'Fold' from the popup menu, then ARivaT will take care of folding the CMP into its requesting CP. References [29] and [30] proposed some modifications to the Riva-based BPA modelling approach to deal with the impact of folding a CMP on Case Strategy Processes. These proposed changes are summarized as follows. When a CMP is folded, then the CSP associated with the CMP to be folded is included into the requesting CP associated CSP. Therefore, the associated CSP of the requesting CP will maintain a strategic view of the flow of cases of the requested CP and preserve a strategic view of the requested CP itself as well. Consequently, a 'guide' interaction is initiated from requesting CSP to the requested CP. We have implemented these adaptations in ARivaT and the outcomes are explained in Figure 8.

The left side of Figure 8 shows a general example of a Second-Cut PAD and the right side of the same figure contains the result of folding 'Manage the flow of UOW2 cases' CMP. The second heuristic is folding a CMP of the part into the CP of the whole. This scenario is totally based on user decisions. Hence, if the user decides to fold a CMP for that reason, then user can follow the same aforementioned steps for folding a CMP. The third heuristic is dealing with delivery interactions and delivery chains. One case to consider is when the user examines if a delivery interaction exists in reality and if not, then the user can right-click on the edge representing the interaction and choose to delete it. Another case to consider is to look for short-circuited delivery chains and check if these chains have resembled reality. In this case, it is a troublesome task for users to detect delivery chains in the Second-Cut PAD, particularly when the size of the diagram is large.

Consequently, ARivaT is designed to automatically discover all deliver chains and list them in the deliver chains drop-down list in the toolbar. If the user selects a deliver chain from the drop-down list, as shown in Table 1, then the selected deliver chain is highlighted in orange as illustrated in Figure 7. After that, the user can click the 'Replace Chain' button to replace the currently selected chain with its short-circuited one. The refresh button next to the 'Replace Chain' button is used to refresh the list of deliver chains in the drop-down list.

Detecting deliver chains is conducted using a variation of the Depth-First traversal of the Second-Cut PAD where only 'Deliver' edges are taken into account and which is explained in Algorithm 2. Line 16 stops the recursion if the length of the adjacency list of a node is zero or the node is visited. Once the if statement in line 21 is satisfied, then a new deliver chain is established and therefore it is inserted into 'globalPathList'. Line 27 removes the first node from 'localPathList' path when there is a cycle. The fourth heuristic is dealing with CMPs resulting from 1..1 interactions coming from UOWs

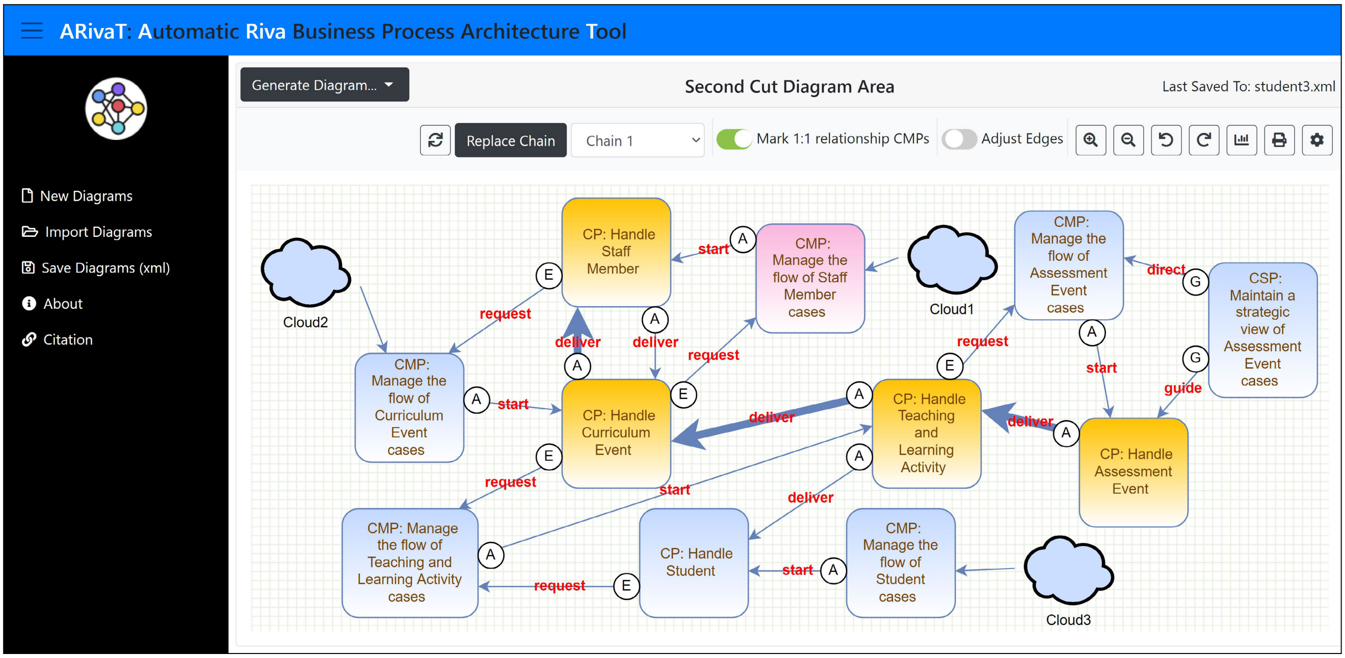


FIGURE 7. Initial draft of Second-Cut PAD (identical to First-Cut PAD) with one of the deliver chains highlighted and CMP's resulting from 1..1 relationships highlighted.

diagram. The user can easily detect these CMPs by turning on the 'Mark 1..1 relationship CMPs' switch as shown in Table 1 which marks the related CMPs in purple and then the user can choose to fold any CMP using the same folding process explained earlier.

Figure 7 shows that 'Manage the flow of staff member cases' CMP is marked in purple as it is resulting from 1..1 interaction in the related UOWs diagram. The fifth heuristic is dealing with empty CMPs, wherein a UOW simply does not need an associating CMP and in this case, the user can choose to fold the empty CMP. Figure 7 shows how the Second-Cut PAD for the higher education process architecture looks like such that one of the deliver chains is selected and highlighted in orange and CMPs resulting from 1..1 interactions are marked in purple. On the one hand, this figure shows the initial draft of the Second-Cut PAD without yet applying any of Ould's heuristics. On the other hand, Figure 9 illustrates the Second-Cut PAD generated in ARivaT after deleting 'student' CMP, folding 'Assessment Event' CMP, and replacing deliver chain 1 with a short-circuited one.

5) SUMMARY

Sub-section II-B highlighted the development methods that we followed in designing ARivaT. The complexity of the tool and its constituent components is recognized but it is also essential to accommodate the early-identified requirements for ARivaT. Furthermore, the user interventions/decisions have a large impact on the final output of ARivaT which is reasonable in the context of business process architecture complexity. For example, ARivaT can fold a CMP on behalf of the user, however, it is the user decision to order ARivaT

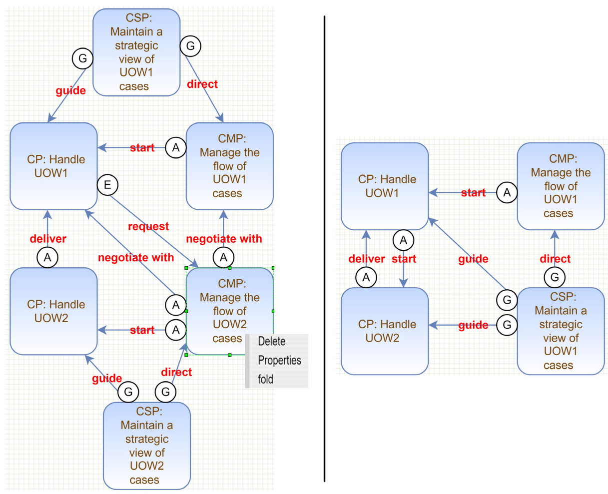


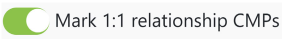
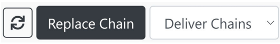
FIGURE 8. Folding a CMP in ARivaT. Result after folding is on the right.

to perform the folding process. Also, the user is the one responsible of deciding to remove or keep a given interaction.

C. ARivaT VALIDATION

This sub-section aims at answering the following question: is ARivaT capable of properly representing a process architecture? To answer the previous question, we modeled the higher education in universities case study which is already been presented and validated by [31]. In this case study, the following essential business entities exist. Student, staff member, curriculum element, teaching and learning activity, and assessment event. All of the previous EBEs are self-explanatory and each of them is turned into a Unit of Work. The interactions

TABLE 1. List of toolbar features specific to Second-Cut PAD.

Toolbar Feature	Explanation
 Mark 1:1 relationship CMPs	A purple color is used to mark all CMPs in the Second-Cut PAD that are involved in 1..1 interactions coming from UOWs diagram
 Replace Chain, Deliver Chains	The drop down list contains all deliver chains found in the Second-Cut PAD. Choosing one of them causes the selected deliver chain to be highlighted in orange. The replace chain button is used to replace the currently selected deliver chain with its corresponding short circuit one. The refresh button is used to refresh the drop down list of deliver chains.

between those UOWs occur as follows. Staff members (faculty members) work on curriculum elements and make them suitable for teaching and learning by proposing the content of the curriculum, and therefore, we can state that curriculum elements are generated by staff members. Looking at this interaction from the opposite side, curriculum elements demand the existence of staff members for teaching purposes. Curriculum elements generate teaching activities for staff members and learning activities for students. Teaching and learning activities require the existence of assessment events which help in the measurement of student achievement on the one hand and the success of staff members on the other hand.

We employed ARivaT to model the previous process architecture and this resulted in the UOWs diagram in Figure 6. After that, we used the automatic generation features of ARivaT to generate the First-Cut PAD and Second-Cut PAD (Figure 7 and Figure 9). Analyzing the outcomes of ARivaT, we concluded that the three diagrams resulting from ARivaT correctly and completely represent the higher education case study and the generated diagrams are identical to the ones found in the case study [31]. However, we state that the degree of correctness and completeness of generating BPA models by ARivaT can be better judged if ARivaT is tested against additional case studies which is part of our future direction.

III. RESULTS

In this section, we describe an experiment that we conducted to prove the effectiveness of ARivaT. In addition, we explain the different charts and statistics that ARivaT is able to generate to assist the modeler with valuable insights about Riva BPA diagrams.

A. ARivaT EVALUATION

To evaluate the effectiveness of ARivaT, a hybrid evaluation approach has been used where two samples are used to ensure

Algorithm 2 Deliver Chains Detection
Input: Graph: mxGraph Model of Second-Cut PAD
Output: List of Deliver Chains Stored in Variable 'globalPathList'

```
function findDeliverChains(graph)
    let vList = graph.getChildVertices();
    let adjList = initAdjList(vList);
    globalPathList = new Array();
    for (each node V_i in vList) do
        let isVisited = new Array(vList.length);
        for (each node V_j in vList) do
            isVisited[V_j.id] = false;
        end for
        let pathList = [];
        pathList.push(V_i.id);
        getAllPathsUtil(V_i, adjList, isVisited, pathList);
    end for
end function
function getAllPathsUtil(u, adjList, isVisited, localPathList)
    if ((adjList[u.id].length == 0) or (isVisited[u.id] == true)) then
        return;
    end if
    isVisited[u.id] = true;
    for (each node V_i in adjList[u.id]) do
        if (isVisited[V_i.id] == false) then
            localPathList.push(V_i.id);
            if (localPathList.length > 2) then
                globalPathList.push(localPathList)
            end if
            getAllPathsUtil(V_i, adjList, isVisited, localPathList);
            Remove first node in localPathList in case of a cycle
        end if
    end for
    isVisited[u.id] = false;
end function
```

the validity of the outcomes of this research. First, a group of 53 students has been selected to evaluate ARivaT via drawing the UOWs diagram in Figure 6 and its corresponding First-Cut PAD in Figure 7. Students were asked to draw the previous diagrams with general-purpose tools (MS Word, MS Visio), specialized tool (Camunda), and ARivaT, and we measured the time it took each student to finish the task. In terms of the time needed to generate the diagrams, this experiment (Figure 10) shows an average time saving of 82 percent when using ARivaT as opposed to using MS Word and MS Visio. Also, an average of 69 percent is witnessed when using ARivaT compared to using Camunda. In both cases, this significant time saving resulted from the fact that none of the previous tools have forward engineering feature that automatically generates First-Cut PAD and Second-Cut PAD from their corresponding UOWs diagram. However, in the case of comparing ARivaT to Camunda, the time saving dipped to 69 percent which is reasonable because specialized tools such as Camunda have specific BPA notation and advanced grid features that makes user's life easier when drawing BPA diagrams. Due to space limitations, only

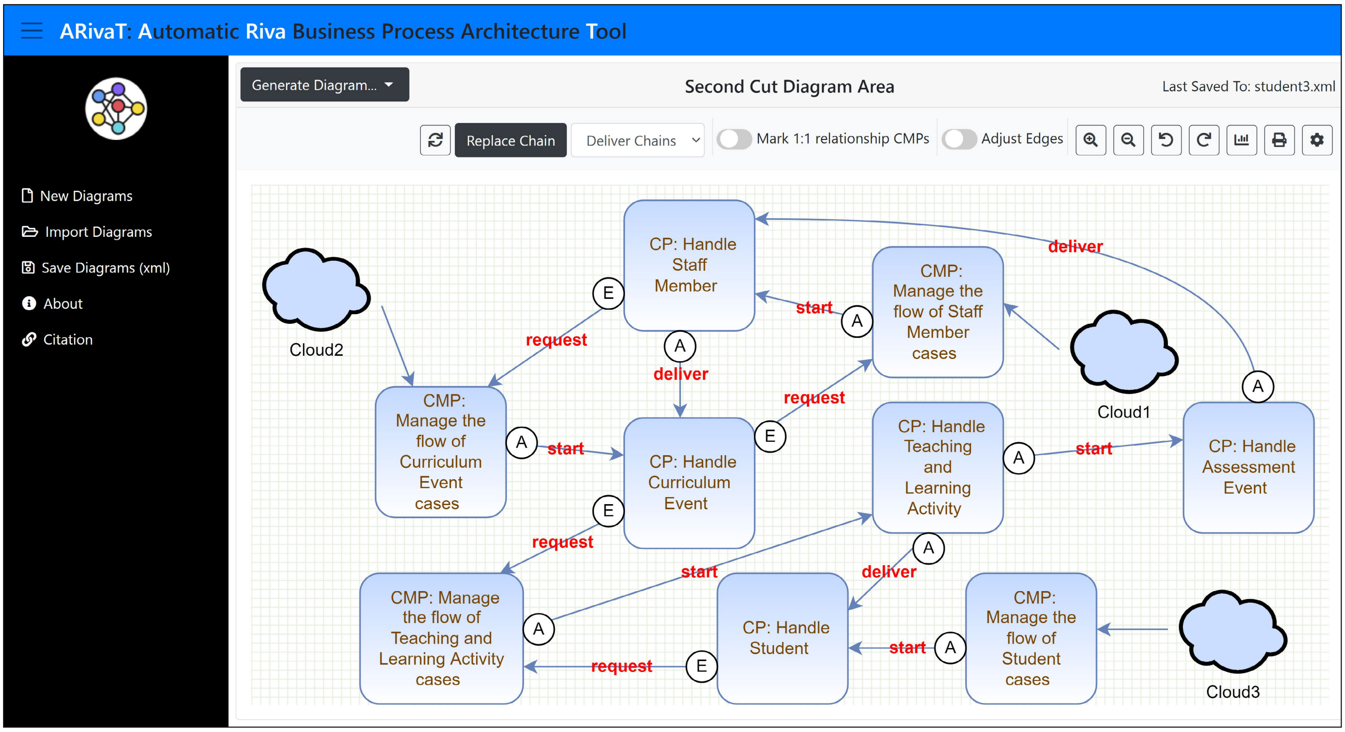


FIGURE 9. Second-Cut PAD after deleting student CMP, folding one CMP, and replacing a deliver chain with a short circuit one.

20 students out of the 53 student sample are shown in Figures 10 and 11.

Moreover, we investigated how complete the user's work is when using MS Word and Camunda as opposed to using ARivaT. As Figure 11 shows, there is a variance of completeness between students' works and this is understandable because students relied on their own skills and their own judgment to decide whether their work is complete. In contrast, ARivaT built-in generation rules increase the degree of completeness of generated diagrams. It is noticed that BPA diagrams generated by Camunda are more complete than the same diagrams generated by MS Word and that falls back to the fact that special BPA notations can be utilized in Camunda which relieves the user from looking into primitive diagram details. Second, a sample of 7 experts who have been teaching software engineering, business process modeling and information systems for more than 3 years has been formed. They have been given the same instructions given to students. The results showed a relatively different but consistent conclusion. The time taken by experts to develop the BPA diagram was limited compared to the time taken by students. Surprisingly, the time taken by experts to use ARivaT was slightly more than the time taken by students due to the experts' willingness to check every single feature in the tool and also reflect on the options given to them.

Overall, the outcomes of both samples, i.e., students and experts, were very close to each other with no significant differences among them. Also, from a statistical perspective, a sample of 53 students and 7 experts might be considered relatively low. However, it is challenging to arrange such experiments with a large number of students and experts due to the complexity of the subject as well as the time frame for this research. Nonetheless, this is an initial evaluation where further evaluation remains for future research direction.

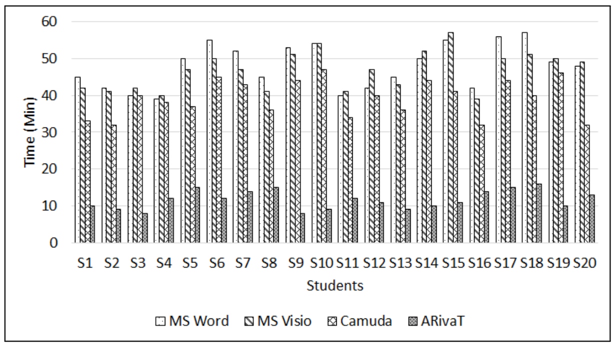


FIGURE 10. Time needed to draw/generate higher education UOWs diagram and First-Cut PAD using ARivaT, MS Word, MS Visio, and Camunda.

We highlight few points of comparison between general-purpose tools, specialized tools, and ARivaT. MS Word and MS Visio have the necessary drawing symbols, however, the notations in Camunda and ARivaT are more Riva specific. Also, all of them possess the capability to save and load diagrams. MS Word and MS Visio are not available as an open source code, however, Camunda and ARivaT are provided as an open source software. Camunda has a special feature which MS Word, MS Visio, and ARivaT don't have

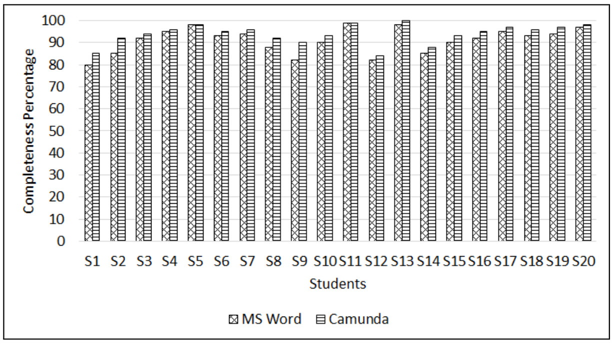


FIGURE 11. Completeness percentage of First-Cut Pad when students draw it in MS Word and Camunda.

which is the ability to allow users to automate BPA execution which assists the user in applying a whole part of BPA without going through each single step. The main advantage of ARivaT over MS Word, MS Visio, and Camunda is still the major time saving when generating Riva BPA diagrams. However, ARivaT is empowered with several unique features that MS Word, MS Visio and Camunda don't have. ARivaT is capable of generating insights from Riva BPA diagrams which helps stakeholders better understand their business workflow. Further, ARivaT enforces Riva rules when generating Riva diagrams which minimizes human errors when creating these diagrams. Furthermore, Riva BPA diagrams can be saved as XML machine readable code which is useful for research development. In addition, detecting some parts of Riva diagrams such as deliver chains can be challenging task in general-purpose tools and specialized tools, however, it is provided seamlessly in ARivaT. ARivaT is more agile in helping users accommodating changes to Riva diagrams such as folding CMP processes easily in one step. The comparison between ARivaT, MS Word/MS Visio, and Camunda is outlined in Table 2.

TABLE 2. Comparison between ARivaT, MS Word/MS Visio, and Camunda.

Feature	ARivaT	MS Word and MS Visio	Camunda
Automatic BPA diagram Generation (Forward Engineering)	Y	X	X
Extracting insights from BPA diagrams	Y	X	X
Availability as Open Source	Y	X	Y
Enforce rules of Riva Business Process Architecture	Y	X	X
Automatic validation for the produced BPA diagrams	Y	X	X
Time and effort saving features	Y	X	Y
Translate BPA diagrams to machine readable code	Y	X	X
Searching for notations inside existing BPA diagrams	Y	X	X
Availability of necessary basic drawing symbols	Y	Y	Y
The utilization of standard notations to ensure the consistency of the produced models	Y	X	Y
Agility as it allows modelers to easily manage their BPA diagrams and accommodates continuous changes	Y	X	Y
Dedicated to Riva models	Y	X	X
Saving and Loading BPA diagrams	Y	Y	Y
Automate process execution	X	X	Y
Cloud version available	X	Y	Y

B. ARivaT BUILT-IN CHARTS AND STATISTICS

ARivaT is capable of generating statistics and charts about the currently displayed Riva diagram. Some of the statistics are common among the three types of diagrams, namely, node count, edges count, and node average degree. For the UOWs diagram, statistics include the number of service interactions and the number of task force interactions. ARivaT provides charts about the currently displayed diagram and sheds light on the features of the diagram. The charts display information per node, some of this information is common between the three types of diagrams, namely, node degree, indegree, and outdegree. Other charts are specific to the UOWs diagram, namely, the number of in-service interactions, number of out-service interactions, number of in-task force interactions, and number of out-task force interactions. Charts can elevate the level of stakeholder understanding of the business. For example, the user can realize which Units of Work and processes are the most vital in the flow of business by looking at the Units of Work and processes that have a higher degree which indicates they are heavily involved in the organization's business.

Nodes with the highest indegree can be an indication that those nodes provide lots of services or perform tasks for other nodes. This implies that these nodes are key factors to the success of the organization's business, and hence they deserve the utmost attention from the stakeholders. Moreover, if a UOW or a process has a high outdegree, then it means that it is heavily dependent on other Units of Work and processes, and hence the stakeholder might think of a way, if possible, to break down this high dependency by for example decomposing the Unit of Work to several other Units of Work. The following figures offer statistics and charts about the diagrams related to the higher education process architecture. Figure 12 provides statistics about the UOWs diagram, namely, nodes count, edges count, average node degree, service interactions count, and task force interactions count. ARivaT can generate similar statistics about First-Cut PAD and Second-Cut PAD except the last two statistics which are specific to the UOWs diagram. In addition, Figure 13 shows statistics about the degree for the nodes in the UOWs diagram. ARivaT is capable of generating related charts representing the indegree and outdegree for UOWs diagram.

Statistics Type	Statistics Value
Nodes Count	8
Edges Count	8
Node Average Degree	2
Service Interations Count	2
Task Force Interactions Count	3

FIGURE 12. UOWs diagram statistics for the higher education process architecture.

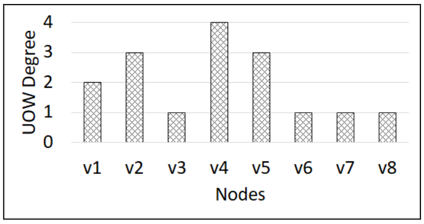


FIGURE 13. UOWs diagram node degree for the higher education process architecture.

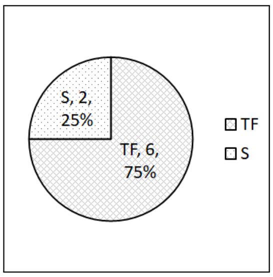


FIGURE 14. Percentage of 'Task Force' interactions and 'Service' interactions in UOWs diagram of the higher education case study.

Likewise, degree, indegree, outdegree for First-Cut PAD and Second-Cut PAD can also be provided by ARivaT. Figure 14 represents the distribution of Task Force and Service interactions in the UOWs diagram and that indicates whether the business flow is service-dependent or task force-dependent. Similarly, Figure 15 depicts the distribution of CP, CMP, and CSP nodes in the First-Cut PAD. Here, it is expected for the percentage of CP and CMP nodes to be equal or close to each other. However, you can tell from that figure how prominent the role of CSP nodes is in the business flow.

IV. DISCUSSION

The previous sections showed that ARivaT capability of saving time, easing the process of generating BPA diagrams via forward engineering, and providing insights about diagrams is evident. However, in this section, we start with answering the research questions that we raised in Section I, then we discuss several additional points that emphasizes the value of ARivaT and mentions ARivaT limitations and outline future research directions.

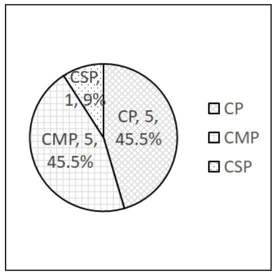


FIGURE 15. Percentage of 'CP,' 'CMP,' 'CSP' nodes in the First-Cut PAD of higher education case study.

During this work, we answered the research question *RQ1* by analyzing candidates of general-purpose tools and specialized tools, and therefore, the answer is no, no current tools are specifically tailored to Riva. By answering *RQ1*, we thought of features that we can employ in ARivaT to make it well suited for Riva BPA modelling. We fulfilled that by furnishing ARivaT with powerful forward engineering capabilities that auto-generates First-Cut PAD and Second-Cut PAD from their corresponding UOWs diagram. This feature led to drastic time reduction in generating Riva BPA diagrams. Also, we empowered ARivaT with the ability to generate charts that provides stakeholders with valuable insights about current Riva BPA models. In addition, we equipped ARivaT with several tasks that can be automatically be performed on the fly such as folding CMP processes. This aids in minimizing the time needed to generate Riva BPA diagrams. Having the previous points in mind, the answer for the research question *RQ2* is yes, having Riva specific tools is useful mainly due to reducing the time needed to generate Riva BPA diagrams.

Other than time reduction features, we discuss several aspects that highlights the value of ARivaT and point out ARivaT limitations. ARivaT can enforce rules of Riva Business Process Architecture in comparison to general-purpose and specialized drawing tools that don't have any rules specific to Riva. Consequently, Riva diagrams generated by ARivaT are automatically validated because they adhere to Riva rules. Using specialized modeling tools assists in detecting syntax errors and inconsistencies in diagrams [32]. In addition, when a tool with standard notation is utilized, the diagrams generated by that tool will be consistent with each other and that aids in the standardization of the drawing style, therefore ARivaT outperforms general-purpose tools that lack these notations. However, the availability of standard notations is something that ARivaT share with specialized tools such as Camunda.

Further, as Riva diagrams become larger and more complex, it is extremely hard for users to manage the diagrams and manually perform some tasks such as marking CMPs involved in 1..1 relationships and detecting deliver chains and replacing them with their corresponding short-circuited ones. ARivaT automates the previous tasks which allow users to handle them easily. Additionally, ARivaT translates Riva diagrams to machine-readable XML code which can be saved/loaded per user desire.

Part of our research interest is sustainability, hence, we show the relationship between ARivaT and sustainability. Process modelling is important in attaining sustainability [22]. According to [33] and [34], sustainability can be categorized into five categories, namely, environmental, economic, individual, social, and technical. ARivaT supports environmental and economic sustainability because it relies on light software and technology layer such as Javascript, which can run on old computers with old specifications. Therefore, old computers don't have to be replaced. Also, all hardware emit carbon, however, ARivaT works locally, so it does not require client/server interaction and this avoids carbon emission by network switching devices. Nevertheless, more efforts are needed to cover other aspects of sustainability such as technical sustainability in ARivaT.

ARivaT has its own limitations. One limitation of ARivaT arises in cases where organization's business processes and business flows are not complex. In this case, the value provided by other tools and ARivaT is comparable. Therefore, ARivaT is not superior over other tools in this case. Also, ARivaT currently works as an offline code which limits user's mobility as opposed to the case when a given software is hosted on a server. Moreover, we validated ARivaT using one case study related to higher education, however, validation on additional use cases would enable us to strengthen the support of ARivaT validity.

As a future direction, we plan to look at technical sustainability such as improving the maintainability of ARivaT code which helps in making ARivaT a more sustainable software able to evolve. In addition, we plan to look into optimizing ARivaT code in a way that conserves the usage of hardware resources which in one hand makes ARivaT more efficient and on the other hand limits carbon emissions by computer hardware. We also aim at developing a cloud based version of ARivaT that increases its mobility. Further, we intend to add a feature to ARivaT which allows user to supply configuration parameters and ARivaT will use these parameters to automate the execution of a Riva BPA diagram. This empowers users with the ability to see the outcome of a given scenario without manually going through each step. Finally, we seek to perform more validation of ARivaT by testing it against additional test cases.

V. CONCLUSION

Business Process Architecture (BPA) helps in understanding an organization's business and that leads to better stakeholders understanding of their role in the workflow of organization's business processes. Riva is an object-based modeling approach for BPA in which three key process architecture diagrams are designed, namely, Units of Work, First-Cut, and Second-Cut. Existing tools either have general drawing capabilities or have limited forward engineering features. In addition, none of the existing tools is specifically tailored to Riva which causes users to miss some drawing details when using these tools and that makes their work incomplete. Therefore, we introduced ARivaT, which is a web-based tool to automate the process of generating Riva BPA diagrams. ARivaT is user-friendly and it also provides statistics and analysis about the drawn diagrams which increases the awareness of stakeholders about their business workflow. In addition, ARivaT draws and generates diagrams in a standard way with attention to detail which assists users in generating elegant diagrams with little effort. Moreover, ARivaT enables users to work progressively on diagrams by allowing them to store/load them as XML files. Furthermore, diagrams generated by ARivaT adhere to the Riva steps and heuristics. We made the tool, along with its instructions, available for the benefit of the research community through Mendeley data [23]. ARivaT was validated by employing it to generate Riva BPA diagrams for the higher education case study. Our experiments showed that using ARivaT to generate Riva diagrams resulted in a considerable 82 percent time reduction when compared to using general-purpose tools such as MS Word and MS Visio. Also, a time saving of 69 percent is recorded when ARivaT is employed as opposed to utilizing special tools such as Camunda. We compared ARivaT with few general-purpose tools and specialized tools and with that regard we marked points of strength and limitations of ARivaT and we also framed our outlook of future work to improve ARivaT.

REFERENCES

[1] P. Senkus, W. Glabiszewski, A. Wysokinska-Senkus, and A. Panka, "Process definitions–critical literature review," *Eur. Res. Stud.*, vol. 24, no. 3, pp. 241–255, 2021.

[2] F. Corradini, F. Fornari, A. Polini, B. Re, F. Tiezzi, and A. Vandin, "A formal approach for the analysis of BPMN collaboration models," *J. Syst. Softw.*, vol. 180, Oct. 2021, Art. no. 111007.

[3] J. Jeston, *Bus. Process Management: Practical Guidelines to Successful Implementations*. Evanston, IL, USA: Routledge, 2018.

[4] M. A. Ould, *Bus. Process Management: A Rigorous Approach*, 1st ed. Swindon, U.K.: British Computer Society, 2005.

[5] L. Cruz, M. Basto, J. Silva, and N. Lopes, "Business process management as a driver for digital transformation : A case study in a higher education institution," in *Proc. 16th Iberian Conf. Inf. Syst. Technol. (CISTI)*, Jun. 2021, pp. 1–6.

[6] S. K. Avezimbetovich, "Improving the strategic management system of automobile enterprises," *Middle Eur. Sci. Bull.*, vol. 20, pp. 112–118, Jan. 2022.

[7] A. Leshob, H. Mili, J. Gonzalez-Huerta, and A. Boubaker, "A value-oriented approach to business process specialization: Principles, proof-of-concept, and validation," *J. Syst. Softw.*, vol. 127, pp. 120–149, May 2017.

[8] A. Anand, S. F. Wamba, and D. Gnanzou, "A literature review on business process management, business process reengineering, and business process innovation," in *Proc. Enterprise Organizational Modeling Simulation*. Berlin, Germany: Springer, 2013, pp. 1–23.

[9] A. Samhan, M. Odeh, M. Kossmann, and A. Tbakhi, "Business process architecture-driven change management applied to a cancer care organization in a systems of systems context," in *Proc. 1st Int. Conf. Cancer Care Informat. (CCI)*, Nov. 2018, pp. 205–214.

[10] A. Samhan, M. Odeh, J. Sa, and M. Kossmann, "OntoSoS.CM: A business process architecture driven and semantically enriched change management framework for systems of systems engineering," in *Proc. IEEE Int. Symp. Syst. Eng. (ISSE)*, Oct. 2016, pp. 1–7.

[11] R. Dijkman, I. Vanderfeesten, and H. A. Reijers, *The Road to a Business Process Architecture: An Overview of Approaches and Their Use*. Eindhoven, The Netherlands: Einhoven Univ. Technol., 2011.

[12] R. Dijkman, I. Vanderfeesten, and H. A. Reijers, "Business process architectures: Overview, comparison and framework," *Enterprise Inf. Syst.*, vol. 10, no. 2, pp. 129–158, Feb. 2016.

[13] R. M. Yousef, "BPAOntoSOA: A semantically enriched framework for deriving SOA candidate software services from Riva-based business process architecture," Ph.D. dissertation, Dept. Comput. Sci. Creative Technol., Fac. Environ. Technol., Univ. West England, Bristol, U.K., 2010.
[14] M. Ahmad, "Semantic derivation of enterprise information architecture from Riva-based business process architecture," Ph.D. dissertation, Dept. Comput. Sci. Creative Technol., Fac. Environ. Technol., University West England, Bristol, U.K., 2015.
[15] P. Harmon, *Business Process Change: A Business Process Management Guide for Managers and Process Professionals*. San Mateo, CA, USA: Morgan Kaufmann, 2019.
[16] A. Samhan, "OntoSoS.BPA.ChM: A semantically-enriched and business process architecture-driven framework for change management in'system of systems' context," Ph.D. dissertation, Dept. Comput. Sci. Creative Technol., Fac. Environ. Technol., Univ. West England, Bristol, U.K., 2020.
[17] M. Ahmad, M. Odeh, and S. Green, "Derivation of a semantic cancer care information architecture from Riva-based business process architecture using the BPAOntoEIA framework," in *Proc. 1st Int. Conf. Cancer Care Informat. (CCI)*, Nov. 2018, pp. 152–164.
[18] M. Odeh, S. Green, M. Kossmann, D. Tbaishat, A. Tbakhi, and A. Mansour, "Evaluating the Riva business process architecture identification method and its constituent heuristics through their application to a cancer care organisation," in *Proc. 1st Int. Conf. Cancer Care Informat. (CCI)*, Nov. 2018, pp. 217–230.
[19] R. M. Yousef and M. Odeh, "The srBPA ontology: Semantic representation of the Riva business process architecture," *Int. J. Comput. Sci. Issues (IJCSI)*, vol. 11, no. 2, p. 84, 2014.
[20] Venic Consulting. *Riva Visio 2003 Stencil, Free Resources From Venice Consulting*. Accessed: Mar. 9, 2022. [Online]. Available: http://www.veniceconsulting.co.uk/veniceresources.htm
[21] C. Inc. *Camunda: Process Orchestration for End-to-End Automation*. Accessed: Apr. 22, 2023. [Online]. Available: https://camunda.com/
[22] R. F. Ismail, F. Safieddine, R. Hammad, and M. H. Kantakji, "Towards sustainable production processes reengineering: Case study at INCOM Egypt," *Sustainability*, vol. 14, no. 11, p. 6564, May 2022.
[23] O. Al-Haj Hassan, A. Samhan, S. AlHajHassan, and R. Hammad, "ARivaT: A tool for automated generation of riva-based business process architecture diagrams," Mendeley Data, V4, 2023, doi: 10.17632/kmhkdd8mkd.4.
[24] I. Beeson, S. Green, and R. Kamm, "Comparative process architectures in two higher education institutions," *Int. J. Organisational Design Eng.*, vol. 3, no. 1, pp. 888–896, 2013.
[25] D. Tbaishat, "Process architecture development using Riva and ARIS: Comparative study," *Bus. Process Manage. J.*, vol. 24, no. 3, pp. 837–858, Jun. 2018.
[26] I. Sommerville, *Engineering Software Products*. London, U.K.: Pearson, 2020.
[27] Jgraph LTD. *Mxgraph*. Accessed: Mar. 9, 2022. [Online]. Available: https://jgraph.github.io/mxgraph/
[28] Jgraph LTD. *Mxgraph Javascript API Specification*. Accessed: Mar. 9, 2022. [Online]. Available: https://jgraph.github.io/mxgraph/docs/js-api/files/index-txt.html
[29] A. Samhan, S. AlHajHassan, and M. Odeh, "Adapting Riva to support case strategy processes and enable modeling a generalized business process architecture," in *Proc. 22nd Int. Arab Conf. Inf. Technol. (ACIT)*, Dec. 2021, pp. 1–6.
[30] A. Samhan and S. AlHajHassan, "An adapted ontology for a generalized riva-driven business process architecture," in *Proc. 22nd Int. Arab Conf. Inf. Technol. (ACIT)*, Dec. 2021, pp. 1–7.
[31] I. Beeson, S. Green, and R. Kamm, "Process architectures in higher education," in *Proc. U.K. Acad. Inf. Syst. Conf.*, 2009, p. 11.
[32] K. Wiegers and J. Beatty, *Software Requirements*. London, U.K.: Pearson, 2013.
[33] C. Becker, R. Chitchyan, L. Duboc, S. Easterbrook, B. Penzenstadler, N. Seyff, and C. C. Venters, "Sustainability design and software: The karlskrona manifesto," in *Proc. IEEE/ACM 37th IEEE Int. Conf. Softw. Eng.*, vol. 2, May 2015, pp. 467–476.
[34] P. Lago, S. A. Koçak, I. Crnkovic, and B. Penzenstadler, "Framing sustainability as a property of software quality," *Commun. ACM*, vol. 58, no. 10, pp. 70–78, Sep. 2015.

OSAMA AL-HAJ HASSAN received the B.Sc. degree in computer science from Princess Sumaya University for Technology, Jordan, in 2002, the master's degree in computer science from the New York Institute of Technology, in 2004, and the Ph.D. degree in computer science from the University of Georgia, USA, in 2010. During his Ph.D. degree, he was with the Distributed Systems Laboratory. He is currently an Associate Professor in computer science. His research interests include web 2.0, mashups, web services, distributed systems, and peer-to-peer networks. He is also a member of the Computer Science Department, Princess Sumaya University for Technology.

AHMAD SAMHAN received the B.Sc. and M.Sc. degrees (Hons.) in computer information systems, and the Ph.D. degree in software engineering from the University of the West of England (UWE), U.K. He is currently an Assistant Professor in software engineering. His Ph.D. degree was about developing a semantically-enriched and business process architecture-driven change management framework in the systems-of-systems context. He was a part of the Software Engineering Research Group (SERG), UWE, during his doctoral studies. His research interests include configuration management, business process architecture, the systems-of-systems, and knowledge-based approaches and ontologies. In addition, he is also a member of the INCOSE Configuration Management Working Group.

SUHAIR ALHAJHASSAN received the bachelor's degree in CIS from the University of Jordan, in 2005, the M.Sc. degree (Hons.) in CIS from Middle East University, Jordan, in 2008, and the Ph.D. degree in software engineering from the University of the West of England (UWE), U.K., in 2019. She is currently an Assistant Professor with the Faculty of Information Technology, Applied Science Private University, Amman, Jordan. She was a Co-Organizer of the yearly MIS Symposium while working with the University of Dammam, Saudi Arabia, in 2013 and 2014. Her research interests include requirements engineering, the systems of systems, business process architecture, and goal-oriented modeling.

RAWAD HAMMAD received the M.Sc. degree in cognitive computing from the Goldsmiths, University of London, in 2010, and the Ph.D. degree in software engineering from the University of the West of England (UWE), in 2018. He is currently the Co-Director of the Smart Health Centre, the Program Leader for the M.Sc. Computing Course and the M.Sc. Digital Education Course, the Technology Enhanced Learning Research Group Leader, and a Senior Lecturer in computer science and digital technologies with the University of East London. He has an extensive experience in software engineering, technology enhanced learning (TEL), artificial intelligence in education, and smart health research and practice. Moreover, he is an Executive Committee Member of the International Society of Artificial Intelligence in Education (AIED) and a Committee Member of different research bodies, and conferences, including EC-TEL, LAK, AI for Post-COVID Education, and Networks in Education.