Visualizing Multiple Process Attributes in one 3D Process Representation

Manuel Gall, Stefanie Rinderle-Ma

University of Vienna, Faculty of Computer Science, Vienna, Austria
`manuel.gall@univie.ac.at, stefanie.rinderle-ma@univie.ac.at`

Abstract. Business process models are usually visualized using 2D representations. However, multiple attributes contained in the models such as time, data, and resources can quickly lead to cluttered and complex representations. To address these challenges, this paper proposes techniques utilizing the 3D space (e.g., visualizing swim lanes as third dimension). All techniques are implemented in a 3D process viewer. On top of showing the feasibility of the proposed techniques, the 3D process viewer served as live demonstration after which 42 participants completed a survey. The survey results support that 3D representations are well-suited to convey information on multiple attributes in business process models.

Keywords: Process Model, Process Attributes, 3D Representation, Virtual Reality

1 Introduction

Companies are sometimes challenged by understanding their own processes as they comprise data from various sources such as databases, services, and real-time information. For their core processes companies use business process models in notations such as BPMN [12]. These models might contain a huge amount of data and can be very complex in their representation. In order to deal with such complex representations, views have proven to be helpful for reducing the complexity by restricting the provided information [15]. However, by using views relations between attributes cannot be identified. To overcome the gap of knowledge representation this paper introduces a 3D representation (3DViz). 3D representations are better suited for presenting more information compared to 2D representations [14]. Some fundamental approaches for 3D processes representations have been proposed [3,4,14]. However, representing 3D processes has not moved as far as other disciplines where 3D representations are common nowadays such as bioinformatics, (3D protein modeling) and mathematics (complex simulations). There are various scenarios where 3D processes representations could be of use like, business process analysis, social modeling, and runtime representation.

In face of those broad applications one might ask why 3D representations are not yet established for business processes? When first works for 3D process representations were released the computers were not able to render huge models

[14]. Furthermore there is a representation gap when visualizing a 3D model on a 2D monitor [14].

However, in the last decade the computing power has increased and therefore rendering huge models is not a challenge any more. In recent literature [6] the second shortcoming (representation gap) is solved by using new devices called head-mounted displays (HMDs) such as virtual reality (VR) and augmented reality (AR) to explore and interact with 3D representations. Figure 1 shows a simple ordering process visualized on a desk containing four *nodes*, a *parallel* and a *loop*. A person wearing a VR device can literally walk around the process. The perspective on the process is changed by an angle of approximately 30 degree per image from ① to ③.
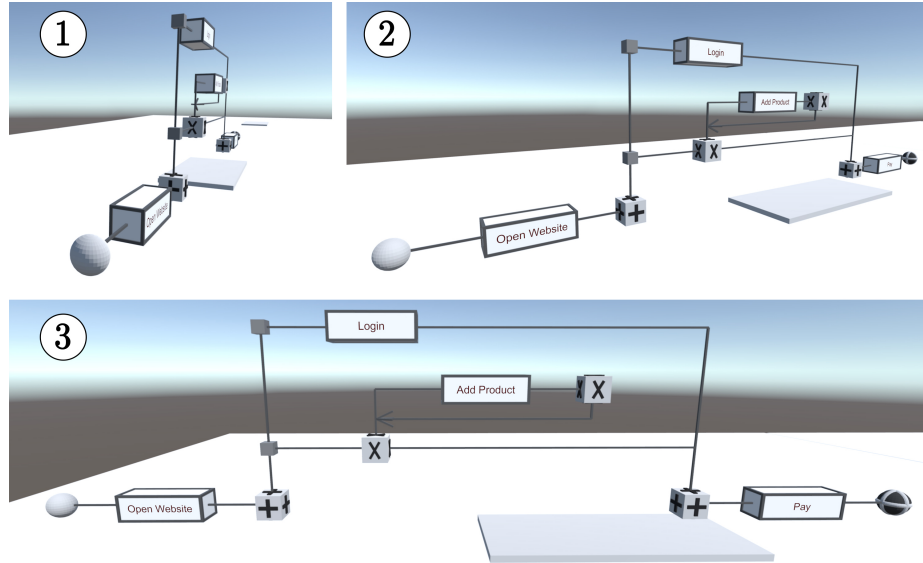

Fig. 1. An order process viewed through a VR device from various angles.

Based on the technological progress (computing power and HMDs) we can argue that the foundation for 3D process representation is available. Within this paper we want to focus on how 3D representations can support the analysis of processes models. While 2D approaches use different views for attribute visualization on a process a 3D representation can contain multiple attributes within one representation.

Specifically we want to look into the following research questions:

RQ1: How can multiple process attributes be integrated in one representation?

RQ2: How can various data types and resources be integrated in a multi-perspective

representation?

RQ3: How to go from a 2D towards a 3D representation?

RQ4: How to improve navigation and orientation within a virtual setting?

These questions impose various challenges, i.e., avoiding cluttered representations, dealing with different data types, and conflicting visual representation.

We use the design science research methodology [17] to answer these questions and tackle the challenges. By using distinct visual styles as well as applying mapping techniques on data types we allow for integrating multiple perspectives into one representation. Section 2 discusses related work for attribute representation and 3D representations. In Section, 3 a fundamental 3D process representation framework (3DViz) is introduced. 3DViz is refined in Section 4 to allow for visualizing multiple attributes within one representation. The created framework is evaluated in Section 5 with a survey and an applicability analysis.

2 Related Work

Within business processes various attributes such as resource, data and time have been identified [1]. Business process modeling languages have been extended to support the creation of individual process attribute visualizations for various stakeholders [13]. Such visualizations are usually limited to represent one process attribute [7,2]. As our contribution is to highlight relations between attributes, approaches that utilize abstraction techniques [15] such as merging activities are not within the scope of the paper.

Besides 2D representations, 3D representations have been used for representing such single attribute process models [3]. Other 3D representation approaches within the business process domain allow for modeling of 3D process [5,4]. When compared to 2D representations such 3D representations allow for incorporating more information into one representation [14]. However, these approaches do not consider representing multiple attributes within one representation.

Additionally these approaches use monitors to visualize 3D representations which is not suitable [14]. Other domains use HMDs to visualize 3D representations [6]. With our research we want to visualize multiple attributes within one 3D representation.

3 Fundamentals of the 3D Approach

Before diving into the contribution we lay out the fundamentals for a 3D process representation framework (3DViz) based on existing work. In this section we discuss the basic representation as well as interaction possibilities with the model.

3.1 Engine

Previous work uses Java3D [14] and Second Life [5] for 3D process representation. However, when compared to how 3D Engines evolved these approaches

seem to be visually outdated. In recent years engines for creating massive 3D worlds went from closed systems accessible for a hand full of developers to broad open communities [16]. There are more then 600 engines currently available [8]. We need an engine that satisfies the following criteria. In order to visualize many instances or big processes the engine has to be able to draw many objects. Furthermore the engine shall allow communication to other services e.g., a process engine. As the proposed framework should not only visualize elements on screen but on other devices as well e.g., on augmented reality glasses and virtual reality glasses, an engine that supports such devices will be selected. Based on these criteria one can see that there have been recent developments in the area of 3D representations e.g., rendering of large amounts of objects, AR, and VR. Therefore we are looking for an engine that constantly aligns to the use of future technology.

Big engines e.g., Amazon Lumberyard, Cry Engine, Godot Engine, Unity and Unreal Engine support most of the criteria. As the authors have previous experience with Unity, Unity is utilized for creating the 3D process representation framework 3DViz.

3.2 Creating a 3D Process Model

In this section every aspect ranging from reading process description files towards visualizing a 3D process model is covered.

File Handling We use XML files from the CPEE [9] process modeling/execution engine as input. These files are not tailored towards a 3D representation as most of the formats out there and this is the challenge we want to tackle. We want to use input files that are not tailored towards 3D to show that those can be used in a 3D representation ($\mapsto$ RQ 3). While reading a file each element e.g., task, loop and parallel is internally stored as an object. In the following we call these objects *Nodes* and store them within a list. A *Node* consists of the following attributes:

- *Types* express the elements control flow behavoiur e.g., loop and parallel.
- *Labels* are used for tasks as well as conditions e.g., XOR conditions.
- *Id's* are unique identifier that allow for *Node* identification.
- *Nodes Before* is a set of *Nodes* that occur immediately before the current *Node*.
- *Nodes After* is a set of *Nodes* that occur immediately after the current *Node*.
- *Arguments* comprise all the additional values stored within the XML file and is designed to be a flexible structure as not all files contain all kinds of arguments. Examples for arguments are the time needed to execute a *Node*, role executing a *Node*, costs when executing a *Node*, services used when the *Node* is executed and many more.

Visual Elements Every type stored within the *Nodes* is represented by one specific 3D visual element. These 3D elements are influenced by BPMN as it is the defacto standard for modeling business processes [12]. Fig. 2 shows all the representation elements in the 3DViz framework containing ① Start Event, ② Task, ③ Parallel Split/Join, ④ Xor Split/Join and Loop, ⑤ End Event, ⑥ Edge and ⑦ Edge with arrow. These visual elements were influenced by BPMN and refined by design principles [10].
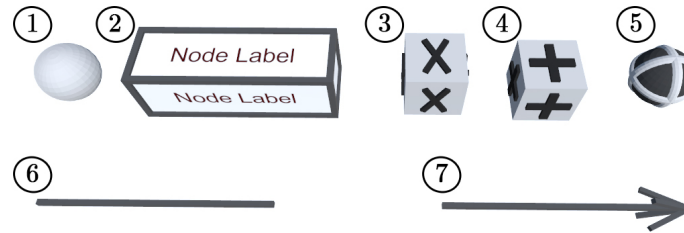

Fig. 2. 3D elements available within the 3DViz framework.

Drawing the Control Flow Creating the control flow is achieved by iterating through a list of *Nodes* and drawing the appropriate visual element determined by the *Type* attribute. There are *Types* leading to a more complex behaviour when drawing the process model e.g., loops, parallel and XOR. When handling these special *Types* we have to be aware that they can influence the positioning of other *Types*. For example, a parallel branch could consist of an XOR with multiple branches therefore the height positioning of the next parallel branch has to be calculated based on the maximum height of the last branch.

Figure 1 depicts an example were the height positioning of the second parallel branch is calculated based on the height of the branch below. This is achieved by drawing the first branch. Then the maximum height of the branch is calculated. Based on the maximum height the second branch is drawn.

3.3 Virtual and Augmented Reality

As literature suggests viewing a 3D model on a monitor might not be the appropriate way to transport the visual fidelity introduced with a 3D model [14]. Therefore we opted for extending our representation beyond desktop monitors. We allow to view a process model with various devices. So far we have implemented the viewing on a monitor and an HTC Vive VR device. Viewing the model in AR is implemented but not tested due to the lack of a device.

3.4 Interaction with the Model

When interacting with a model there are common actions every representation shall support. These are *zooming*, *panning*, *rotating* and *click* actions [14]. Based on the device used for viewing a process model we implemented different interaction possibilities.

Monitor When a monitor is used a mouse interacts with the process model. We allow for *zooming* by using the scroll wheel, *panning* by holding the left mouse button and moving the mouse, *rotating* by holding the right mouse button and moving the mouse and *click* actions by left clicking on a *Node*. Click actions are used to get more details on a specific *Node* e.g., visualizing id, responsible roles and other arguments.

VR Device A VR device allows for the same interactions as the monitor/mouse combination. However, these interactions are implemented in a different way. *Zooming*, *panning* and *rotating* are direct results of the head movement observed by the VR device. For example, when a certain *Node* is interesting leaning towards the *Node* will *zoom* in. The *click* action is realized by using VR controllers. When a controller is near a *node* a *click* selects the node and shows the same information as a mouse click will show.

3.5 Using a 3D Scene

All the figures so far have something in common, they highlight the process by only showing process relevant information. However, within the framework the representation is different.

We digitalized one of our conference rooms. To be as accurate as possible, every objects real world size was measured. Afterwards all the objects were created with a 3D modelling tool. This approach allowed to have a 1:1 scale between real and virtual world. In fact if the real and virtual world objects are aligned the same and an object is touched in the virtual world one would feel the real world object. From a pure visual point of view the created scene can be seen as a digital twin of our conference room. The focus of the scene is on top of two tables in the middle of the conference room. On top of these tables the 3D process models are visualized. Fig. 3 visualizes the real world conference room on the left and the virtual conference room with a process model on the right.

Using a 3D scene aids the navigation and orientation in a virtual environment ($\mapsto$ RQ 4). For example, when loosing focus of the model by moving the head or rotating with the mouse one still knows where to find the model as there are points for orientation within the scene. Such a scene can be slightly compared to a grid or ledger lines within a 2D representation.

We suggest to use scenes for all 3D representations. Scenes can be used in single user settings such as business process analysis and runtime evaluation. In a multi user setting like collaborative modeling scenes can help to improve

communication. We want to point out that a scene around the process only makes sense when wearing a VR device or using a monitor.

Fig. 3. The left side depicts a picture of a real world room, the right side a virtual conference room.

In conclusion these fundamentals allow us to draw a 3D process model read from a file. This model can be explored with various devices and allows some interactions.

4 Attribute Visualization

So far 3D representations have been mainly concerned with control flow visualization. The 3DViz representation presented in this paper extends the control flow view (basic 3DViz control flow elements are presented in Section 3.2) by visualizing multiple attributes within one representation. This particularly enables the detection of relationships between attributes.

4.1 Example Process

For the remainder of the paper we will use the process depicted in Fig. 4. The process describes a common scenario from the care domain for non stationary patients. A patients blood is analyzed to detect a possible disease the patient is suffering from. Fig. 4 shows the process containing these elements: two roles *Nurse* and *Doctor*, six *Nodes*, two *IT-Services* and four data elements *Duration*, *Role Duration*, *Cost* and *Location*. Every *Nodes* is connected to each of the

4 data elements. However in order to keep a readable figure those *Nodes* and data elements are not connected in Fig. 4 as the edges would span across other *Nodes*. The data elements possess different units and data types *Duration* and *Role Duration* in minutes and *Cost* in € are numerical data types while *Location* is a string. Some tasks are linked to *IT-Services*. These services are identified by their URL and represented as string. A *Node* must not contain all the available arguments. For example, the *Node Centrifugation* includes the arguments for all four data elements, the role *Nurse* and no *IT-Service*. Table 1 comprises the provided values for the data elements. These are estimates and shall support the understanding of the visuals shown in the next sections.
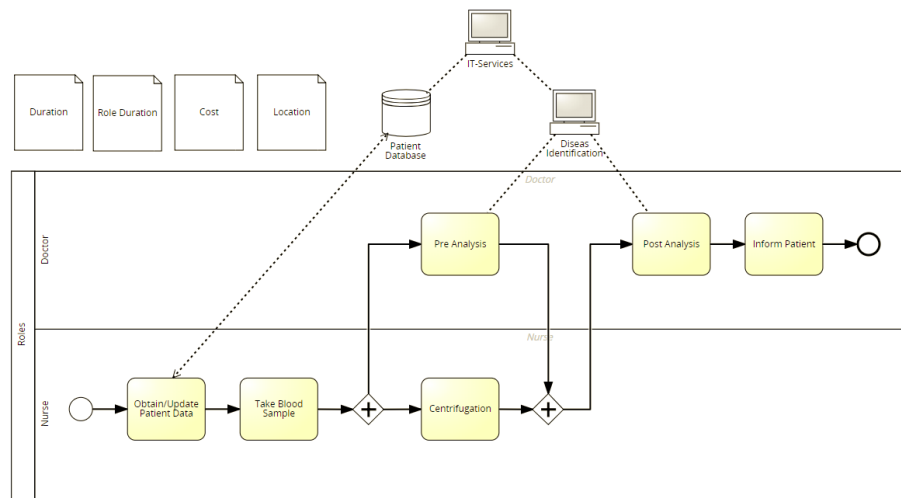

Fig. 4. An example process created with *Signavio* represented as BPMN model and containing different attributes.

	Obtain/Update Patient Data	Take Blood Sample	Pre Analysis	Centri-fugation	Post Analysis	Inform Patient
Duration	20	5	15	10	45	20
Role Duration	1	5	15	1	45	20
Cost	1	10	40	1	90	40
Location	Waiting Room	Treatment Room	Laboratory	Laboratory	Laboratory	Consulting Room
IT Services	Patient Database		Diseas Identification		Diseas Identification	

Table 1. Data element values

4.2   Visual Styles

Within the 3DViz framework three different visual styles are applied. The first is *positioning* of *Nodes* the second *scaling* of *Nodes* and the third *labeling* of *Nodes*. Each *Node* is positioned on the X,Y and Z axis. This allows for integrating more information into the process model representation. The second style uses scaling on *Nodes* and can as well be applied on the X,Y and Z axis. The third approach uses the fact that no longer a rectangle is drawn but a cube. Each side of the cube can be labeled with its own text.For example, Fig. 8 ③ shows the *Name* of a *Node* on the front and the *ID* on top of the cube.

Applying all combinations of the styles *scaling* and *positioning* on a process with control-flow perspective allows us to integrate five attributes within one process ($\mapsto$ RQ 1) plus additional information as a result of the cubes *labels*. In the case a *Node* does not contain a certain data element the *Nodes* style in terms of *scaling* and *positioning* will be set to a default value. To deal with conflicting visual styles i.e. representing *Cost* and *Role* on the Z-Axis, a UI was created to resolve such conflicts manually. The UI allows for one argument per visual style. Arguments are not limited to be visualized by one style. The 3D Viz framework allows for selecting one argument for multiple styles for example, *Cost* could be used on Y and Z Axis scaling. Fig. 5 shows the UI with many filled in values. Every row can be seen as a tuple containing a visual style, the attribute (argument) and a mapping. The mapping describes how the argument is prepared in order to be visualized. Mappings will be covered in more detail in the next section.

In the case one tuple is selected the resulting process model shows a single attribute representation. Fig. 7 ② depicts the example process with only using the tuple from the second row of Fig. 5 visual style *position Z-Axis*, attribute *Role* and *Discrete* mapping. In the case more than one tuple is filled out we call this a multi attribute representation. Fig. 8 depcits a 3D process model containing all the tuples from Fig. 5.

4.3   Data to Visualize

So far 3DViz consists of multiple *Nodes* connected by edges visualizing the control flow view. When modeling a process more information is integrated into the model. Each *Node* can have multiple arguments for example, roles, resources and data elements. These arguments need to be processed before they can be used within the representation. We differentiate 3 processing strategies, *Direct Mapping*, *Relative Mapping* and *Discrete Mapping* ($\mapsto$ RQ 2).

Figure 6 depicts examples for all combinations of visual styles, mappings and arguments. To simplify the figure only the first two *Nodes Obtain/Update Patient Data* and *Take Blood Sample* from the example process are visualized. Further the styles *position Y*, *scale X* and *scale Y* are not depicted as they are similar to the provided styles.

Creating a mapping for arguments imposes certain challenges such as how to handle arguments with non numerical values. Such arguments can not be used for

Fig. 5. User interface for node representation.

calculating minimum and maximum values which is necessary for the *Relative Mapping*. We allow the following mappings for data elements that consist of numerical values.

- *Direct Mapping* ①, ② uses the values of the arguments without further modification. For example, if a value is 5 a *Node* gets scaled 5 times. This mapping might not be of use in many application scenarios, but it can help to detect outliers.
- *Relative Mapping* ③, ④ calculates a percentage value for each *Node* where the minimum is 0% and maximum is 100%. For example, a *Node* with a calculated value of 20% is scaled up by 20%. Up scaling is used instead of down scaling to prevent a node getting invisible when scaled down by 100%.
- *Discrete Mapping* for numerical values ⑤ use multiple swim lanes and interpolate each argument to fit a specific swim lane. For scaling ⑥ each discrete group is represented by its own scaling factor ranging from 0% to 100%.

When the argument to represent is a non numerical value like resources and roles the 3DViz framework does not allow for using *Direct Mapping* and *Relative Mapping*. Such values can only be used with the *Discrete Mapping*.

- *Discrete Mapping* for non numerical values ⑦ represents the arguments values in distinct swim lanes similar to numerical swim lanes. Scaling discrete

non numerical values ⑧ is done by using a scaling factor ranging from 0% to 100% per group
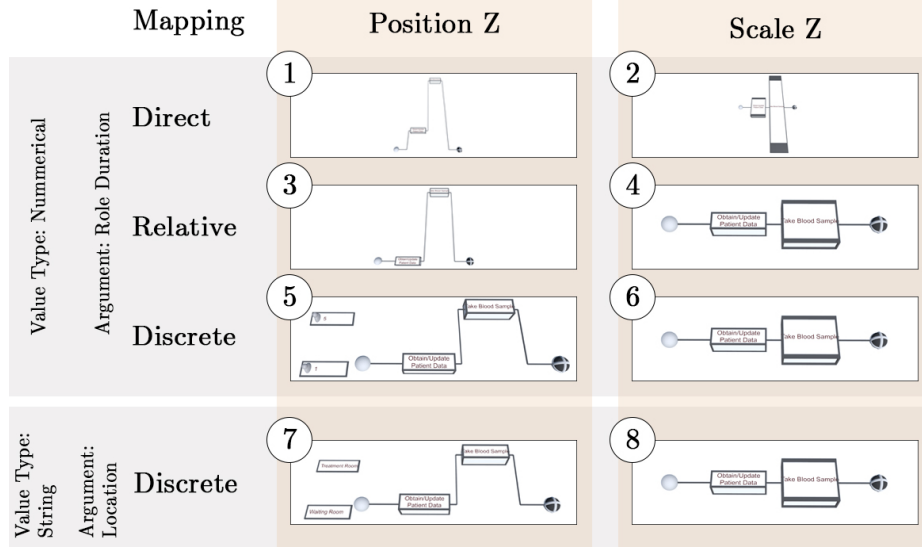

Fig. 6. Mappings applied on different argument types with different styles.

4.4 Representing Single and Multiple Attributes in 3DViz

Visual styles i.e. positioning and scaling are important when one representation contains multiple attributes. These visual styles allow to express each attribute in a unique way. When integrating multiple attributes a unique representation for each attribute is important to be able to distinguish one attribute from another.

Single Attribute Visualization Single attribute representation extends the control flow with one attribute. In case the attribute uses a data-element this element is connected to at least one task. Within the 3DViz framework the tasks are not connected to the arguments. The arguments values are used for positioning and scaling. Omitting the edges for data elements leads to a reduction of visual clutter. This will be even more important when dealing with multiple attributes.

Figure 7 shows the same process with *Signavio Role* attribute on top ①. The 3DViz visualization with style *Position Z, Discrete Mapping*, for the attribute *Role* in the middle ②. By showing one attribute the 3DViz Framework does not convey its full potential. In example ② the angle from top was chosen to show how similar those two representations are.

Figure 7 ③ depicts another single attribute. Representing the style *Position Z, Relative Mapping* for the attribute *Cost*. This approach differentiates from other approaches suggested by literature [7,2] as the arguments values changes the *Nodes* Z-Axis position. One can spot that the *Nodes Pre Analysis* and *Post Analysis* are positioned in the back as those are the *Nodes* with highest cost.
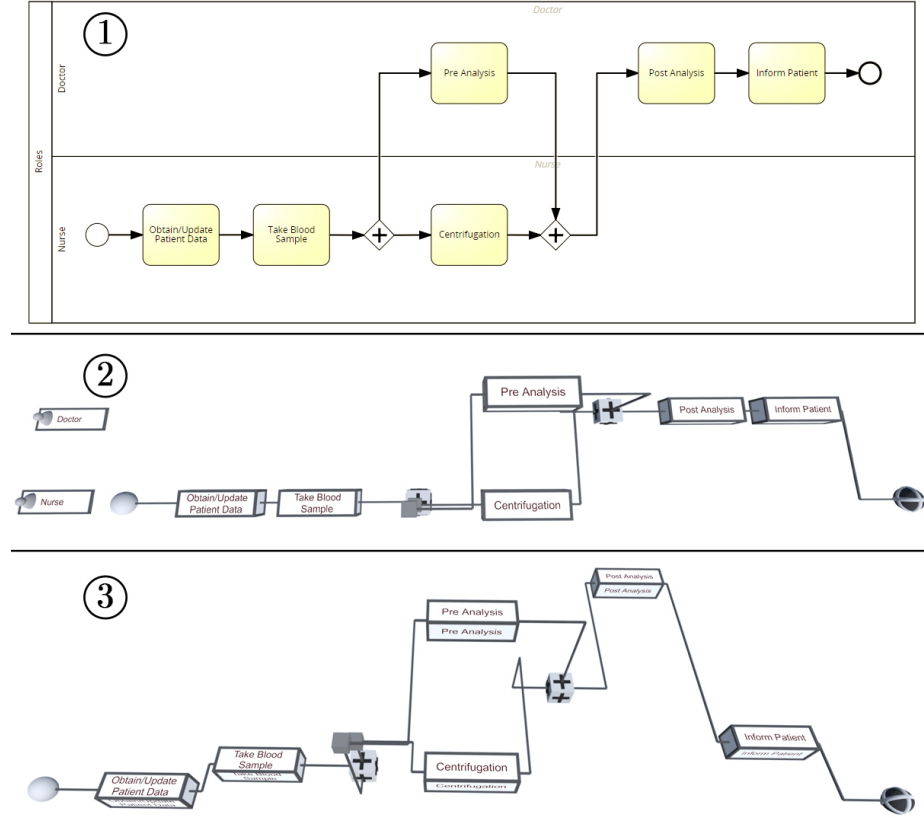

Fig. 7. ① , ② show the argument *role* with *Direct Mapping* on the *Z-Axis.* ③ shows the *Cost* with *Relative Mapping* on the *Z-Axis.*

Multiple Attribute Visualization Limiting one representation to a single attribute aggravates the ability to explore connections between attributes. Shifting from one attribute to another will not allow to recognize these connections either. With 3DViz we allow for combining multiple attributes into one representation (↦ RQ 1). Within a 2D setting comparing arguments can be accomplished by switching between representations. In a 3D representation the arguments can

be included within one representation and compared i.e. by the size of a cube. Further differences when comparing within a 3D representation compared to a 2D representation is that one does not lose focus on the whole picture of the process.

When using multiple attributes we suggest to use non numerical arguments e.g., resources and roles with a *Discrete Mapping* for the positioning on the Y or Z-Axis. While numerical values should be used for scaling of nodes.

Figure 8 shows the example process with the mappings from Fig. 5, *positioning* for *location* ①, and *roles* ②, *scaling* ③ for *duration*, *role duration* and *cost* and *labels* for *id*, *name* and *IT-Service*. On the bottom of Fig. 8 ④ gives an overview of the whole process with the scaling axis depicted on the bottom left. Within the framework the scaling axis is depicted every time a scaling is active.

Based on the 3D representation depicted in Fig. 8 the following conclusions can be made and can be part of an in depth analysis for potential process improvements. Two *Nodes Pre Analysis* and *Centrifuge* are performed in parallel at the same location by two different *Roles* ③. The duration of a *node* is not responsible for a *Nodes Cost* ④Ⓐ,Ⓑ whereas, the *Role duration* and *Cost* seem to be related ③ e.g. *Nodes Pre Analysis* and *Post Analysis*. Under consideration of the arguments *Cost* and *Role duration Doctors* are more expensive than *Nurses*.

The example combines five attributes in one representation. For inexperienced users this can be overwhelming. However, the example demonstrates the full potential of 3DViz. The comprehension one can obtain from a 3DViz representation depends on the complexity of the process model that is visualized and the level of experience a user has with 3D representations.

In conclusion we want to emphasize that representing multiple attributes within one model is a suitable approach to detect relationships between attributes.

5   Evaluation

The evaluation consists of two parts: a user survey and an evaluation on the applicability of the 3DViz framework. The user survey outlines how a 3D process representation is perceived and where the strengths of such a representation can be found. To show the applicability of the 3DViz framework the framework was implemented and a performance analysis carried out.

5.1   User Survey

The first goal of the survey was to evaluate if 3D representations are applicable for business process representation. The second goal was to identify the strengths of a 3D process visualization.
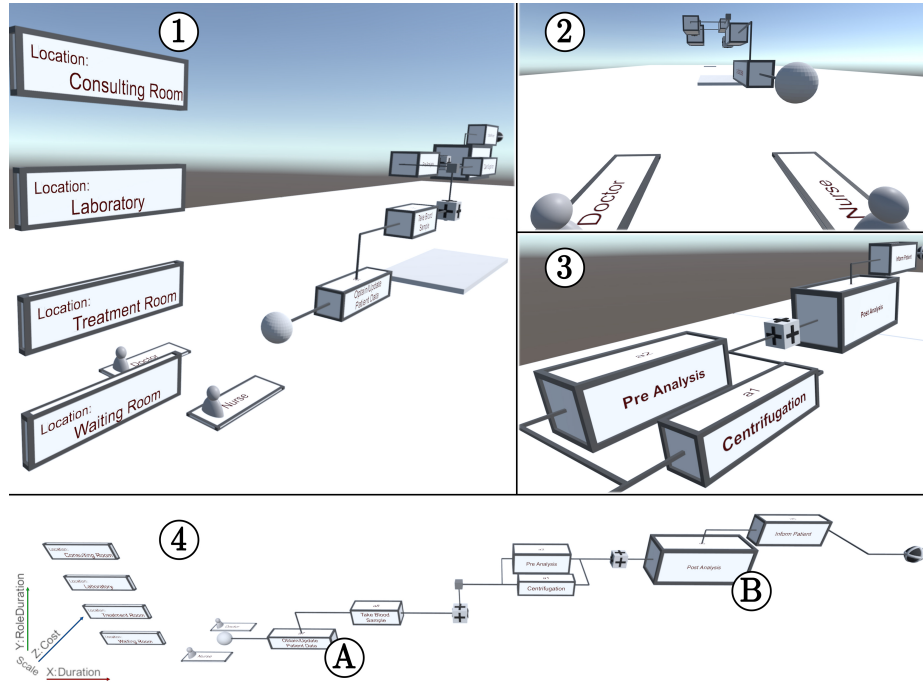
Fig. 8. Different angles on the 3D representation of the centrifuge process.

Survey Design The survey was designed to be conducted during a public event called *Long Night of Science*. During the event the attendees participated in a demonstration of a physical process were goods had to be moved from the storage unit to a consumer. During the execution of the process the attendees were introduced to 3DViz. While they executed the physical process they were able to see and explore the 3D process model.

Based on this experience the participants answered a short survey containing two demographic questions and six 3DViz related questions. The survey was designed in accordance with the *Ten Commandments* [11]. A pretest with four participants, two familiar with the topic and two not familiar with the topic was used to evaluate and improve the question design.

Sample By reaching out during a public event we wanted to reach a diverse set of people. In total 42 participants answered the survey. Figure 9 depicts the distribution for age and education among the participants. The age distributes as followed, 10 (23.8%) participants are older than 63 years, 3 (7.1%) participants between 41-62 years, 24 (57.1%) participants between 23-40 years, and 5 (11.9%) participants between 13-22 years. None of the participants was younger than 13 years. The education is biased towards higher educations such as vocational schools with higher education and university degree each selected by 19

(45.2%) participants. The remaining 9.6% distribute to apprenticeship 3 (7.2%) and elementary school 1 (2.4%).
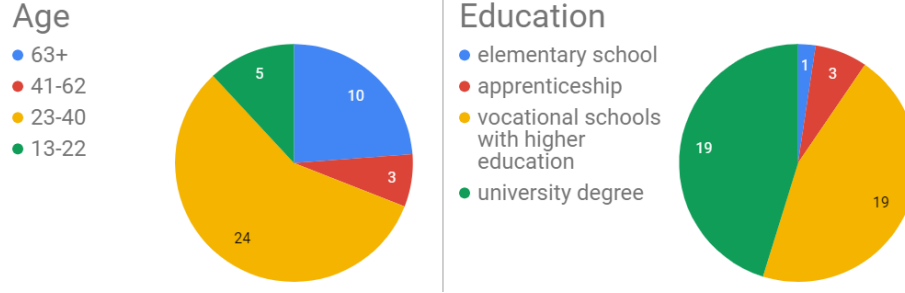

Fig. 9. Age and education of the survey participants.

Results and Discussion The first question the participants had to answer was if they think 3D representations can be useful when representing a process model. In total 38 (92.8%) of the participants answered with yes while 3 (7.2%) answered with no. This indicates a strong sign that 3D representations can be a benefit for process model visualization.

The next question should reveal the areas a participant identifies potential for 3D representations. The survey provided four check boxes and one open form to fill out. Figure 10 summarizes the results. From the 42 participants 33 (80.5%) marked that combining multiple attributes into one representation is a good application for 3D process models. 31 (75.6%) participants found that changing the viewing angle and rotating around the process model is suitable for 3D applications. 25 (61%) of the participants think that 3D representations can be used to express big data. Surprisingly only 11 (26.8%) think that 3D representations are a good fit to represent less overlapping edges. Three Participants contributed to the open question. For general readability the statements are translated from German to English. The first two statements "Use VR for exploration" and "Use AR and VR technology" are covered within this paper. In our demo show case we did not have the possibility to demonstrate 3DViz with such devices. The third statement "Use visually different paths through the process." was so far not under consideration for the 3DViz framework. However, we argue that for the current implementation differentiation between paths is not necessary. Such visually different paths could be interesting when considering process mining or run-time analysis.

The last question of the survey was if using a 3D scene is a good solution to increase orientation within a 3D representation. Answered by 40 participants (97.6%) with yes it is clear that using a 3D scene is a very popular choice.

In summary we identified that 3D representations are suitable for process representations. Within a 3D representation there are areas such as attribute
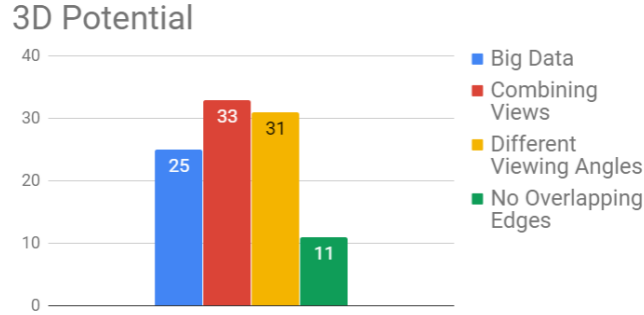
Fig. 10. Potential areas where 3D process visualization can thrive.

combination that can benefit from utilizing the third dimension. When representing a 3D process model a scene around the process was very well received by the participants. Such a scene aids navigation and orientation within a 3D representation.

5.2 Applicability

To demonstrate the applicability of 3D process attribute representation the approach has been prototypically implemented with the Unity framework. The prototype is called 3DViz and available here[1].

We randomly created process models with a tool available here[1]. The tool allows to specify the amount of *Nodes* that shall be created, the amount of arguments and the amount of control-flow related elements such as *Parallel*, *Xor*, and *Loops*. The arguments and control-flow related elements are randomly assigned. Figure 11 ① depicts an example with 4 *Nodes*, 2 arguments, and 2 control-flow elements as input parameters. One of the arguments is applied with discrete mapping on the Z-Axis position the other one with relative mapping on the Z-Axis scaling. Resulting in a process model starting with a *Node* followed by a *Loop* that contains 3 *Nodes* and an *Xor* one of the *Nodes* consists of a different argument and is represented on an other swim lane. For demonstration of a large process Fig. 11 ② depicts an example with 1024 *Nodes*, 512 control-flow elements and 5 arguments one applied per visual style.

For the performance analysis the time between receiving the file from the random process creation tool and drawing of the process is finished is measured. The analysis is carried out on a mid to high end PC with quad-core processor with 3.60 GHz, 16 GB RAM, and a GTX 1060 graphics card. Figure 12 depicts the measurements ranging from 2 *Nodes* and 1 control-flow element depicted as *2N, 1C* up to 1024 *Nodes* and 512 control flow elements *1024N, 512C*. All of the measurements are performed with five arguments. Per group 10 measurements were taken. The average time needed is displayed on the Y-Axis of Figure 12
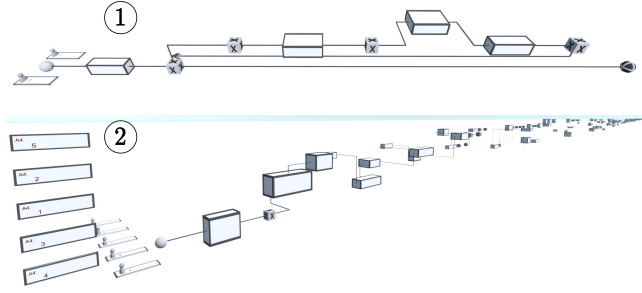
---

[1] http://gruppe.wst.univie.ac.at/projects/crisp/index.php?t=visualization

Fig. 11. Two randomly created process models the first with 4 *Nodes*, 2 control-flow elements, and 2 arguments, the second with 1024 *Nodes*, 512 control-flow elements and 5 arguments. (label unreadability intended)

with the co-efficient of variation displayed on top of each bar. The co-efficient of variation is decreasing from 15% *2N,1C* towards 3.5% for *1024N, 512C*. Figure 12 shows a linear growth demonstrating that the framework is applicable for small and huge process models.

Based on the evaluation results we argue that the applicability is given. 3DViz is capable of displaying models containing 1024 *Nodes* and 512 control-flow elements within 2 seconds.
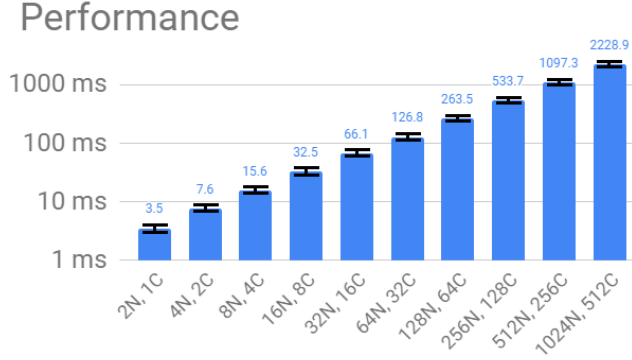

Fig. 12. Perfromance analysis of 3DViz. X-Axis depicts *Nodes* and control-flow elements. Y-Axis depicts the time in miliseconds needed to visualize the process model.

6 Conclusion and Outlook

The proposed 3DViz framework allows to represent multiple attributes such as resources, data, and roles within one 3D representation. Representing multiple

attributes at once allows for finding relations between attributes. Such a representation shows a lot of information therefore we propose to use modern technology like head mounted displays to explore 3D representations. When wearing such a device orientation can be difficult. Therefore we propose to use a 3D scene to aid orientation when exploring a 3D process model.

3DViz was evaluated with a user study and applicability evaluation. The studies show that 3D process attribute representation is a good solution for representing multiple attributes. Performance wise the evaluation showed a linear growth and that the framework is capable of representing huge process models within seconds.

For future work we want to expand towards representing runtime-execution and compliance violations within 3DViz.

References

1. Van der Aalst, W.M.: Process discovery: An introduction. In: Process Mining, pp. 125–156. Springer (2011)
2. Becker, J., Rosemann, M., von Uthmann, C.: Guidelines of Business Process Modeling, pp. 30–49. Springer Berlin Heidelberg (2000)
3. Betz, S., Eichhorn, D., Hickl, S., Klink, S., Koschmider, A., Li, Y., Oberweis, A., Trunko, R.: 3d representation of business process models. MobIS 8, 73–87 (2008)
4. Brown, R.A.: Conceptual modelling in 3d virtual worlds for process communication. In: Proceedings of the Seventh Asia-Pacific Conference on Conceptual Modelling-Volume 110. pp. 25–32 (2010)
5. Brown, R.A., Recker, J.C., West, S.: Using virtual worlds for collaborative business process modeling. Journal of Business Process Management 17(3), 546–564 (2011)
6. Butscher, S., Hubenschmid, S., Müller, J., Fuchs, J., Reiterer, H.: Clusters, trends, and outliers: How immersive technologies can facilitate the collaborative analysis of multidimensional data. In: Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems. p. 90. ACM (2018)
7. Finkelstein, A., Kramer, J., Nuseibeh, B., Finkelstein, L., Goedicke, M.: Viewpoints: A framework for integrating multiple perspectives in system development. Intl. Journal of Software Engineering and Knowledge Engineering 2, 31–57 (1992)
8. Lewis, M., Jacobson, J.: Game engines in scientific research - introduction. Communications of the ACM 45, 27–31 (01 2002)
9. Mangler, J., Rinderle-Ma, S.: Cpee - cloud process exection engine. In: Int'l Conference on Business Process Management. CEUR-WS.org (September 2014)
10. Moody, D.L.: The physics of notations: Toward a scientific basis for constructing visual notations in software engineering. IEEE Transactions on Software Engineering 35 (2009)
11. Porst, R.: Fragebogen. Ein Arbeitsbuch 3 (2011)
12. Recker, J.: Opportunities and constraints: the current struggle with bpmn. Business Process Management Journal 16(1), 181–201 (2010)
13. Rosa, M.L., Dumas, M., ter Hofstede, A.H., Mendling, J.: Configurable multi-perspective business process models. Information Systems 36(2), 313 – 340 (2011)
14. Schönhage, B., van Ballegooij, A., Elliëns, A.: 3d gadgets for business process visualization a case study. In: Proceedings of the Fifth Symposium on Virtual Reality Modeling Language. pp. 131–138. VRML '00, ACM (2000)

15. Smirnov, S., Reijers, H.A., Weske, M., Nugteren, T.: Business process model abstraction: a definition, catalog, and survey. Distributed and Parallel Databases 30(1), 63–99 (2012)
16. Smith, S.P., Trenholme, D.: Rapid prototyping a virtual fire drill environment using computer game technology. Fire Safety Journal 44(4), 559 – 569 (2009)
17. Wieringa, R.: Design Science Methodology for Information Systems and Software Engineering. Springer (2015)