VR-PM: Immersive Process Mining and Process Modeling with BPMN and Petri Nets in Virtual Reality

Roy Oberhauser[0000-0002-7606-8226]
Computer Science Dept.
Aalen University
Aalen, Germany
e-mail: roy.oberhauser@hs-aalen.de

Abstract - **Automation of business and industrial processes are increasing essential for and affect a larger set of stakeholders. Process Mining (PM) relies on execution logs to provide process-centric analysis data, which can support process stakeholders with evidence-based insights and decision support. Yet PM has typically been the purview of specialists, while analysis, insights, and models have not been readily accessible to a larger set of affected stakeholders. This paper contributes VR-PM, an immersive solution concept towards visualizing and interacting with process models (BPMN and Petri Nets), process variants, process conformance checking, and PM data in Virtual Reality (VR). Our realization shows its feasibility, and a case-based evaluation provides insights into its capabilities.**

Keywords - process mining; process analysis; virtual reality; business process management; process conformance checking; process discovery; Petri Nets; BPMN; process modeling.

I. INTRODUCTION

This article extends our paper on VR-ProcessMine [1], adding immersive Petri Net models, Business Process Model and Notation (BPMN) models, process discovery, an alternative process variant visualization, and conformance checking capabilities in VR.

The digital transformation sweeping through society affects businesses and organizations, resulting in an increased emphasis on business agility and automation that affects a larger set of stakeholders. Business Processes (BPs) (or workflows) are one significant automation area, as evidenced by the Business Process Management (BPM) market, which is forecast to grow from $20B in 2024 to $60B by 2030 [2]. Each execution of such a process leaves a digital footprint of process-related events and the timepoint of execution, typically contained in various log files across the various IT systems (business, manufacturing, etc.) involved in an enterprise. BPs are a way for ordering the activities involved in and executed in an enterprise, be they automated, semi-automated, or human-driven, and thus BPM is where much of the value generated by an enterprise is achieved.

Repeatable processes are fundamental for describing and understanding how businesses and organizations operate, and are utilized in IT, manufacturing, Industry 4.0, etc. Increasing digitalization and automation necessitate evidence-based comprehension and analysis of the processes involved, including their variations, anomalies, performance, and evolution. Process Mining (PM) [3] is a sub-field of data science specifically focused on analyzing event data generated when (business) processes are executed [4]. Because PM relies on event logs of actual process executions, it is evidence-based (or fact-based). This analysis can provide essential insights for understanding and optimizing (business) process execution. When referring to processes we assume BPs to be a subset of the more abstract term and will use both terms interchangeably. One process variant represents a set of process instances that resulted in the same sequence of events.

The Process Mining Manifesto [5] describes six guiding principles and eleven challenges. Our contribution intends, in particular, to support these principles:

- *GP1: Event Data Should Be Treated as First-Class Citizens*,
- *GP4: Events Should Be Related to Model Elements*, and
- *GP6: Process Mining Should Be a Continuous Process*.

Furthermore, the contribution seeks to address these challenges in particular:

- *C10: Improving Usability for Non-experts*, and
- *C11: Improving Understandability for Non-experts*, are a primary motivation for our work.
- A secondary effect that is enabled is to support *C9: Combining Process Mining with other Types of Analysis*.

In general, visualization remains a challenge when dealing with large data sets that involve relations and different variation sets. As data and processes become more relevant to the digital enterprise and stakeholders more digitally savvy, it is all the more relevant and challenging to integrate non-expert enterprise stakeholders in process analysis. By leveraging the immersive capabilities of Virtual Reality (VR), BP analysis can be made more accessible to a wider set of stakeholders, such that not just process modeling specialists, but also those directly involved in executing a BP or observing an automated BP can view und gain insights to various issues regarding a BP of interest, including the combination with other relevant enterprise models.

In prior work, we developed VR-BPMN [6] to visualize business processes in VR based on the BPMN notation. Our VR-EA [7] contributed a VR solution for visualizing, navigating, annotating, and interacting with ArchiMate Enterprise Architecture (EA) models. VR-EAT [8] presented our VR-based solution for visualizing dynamically-generated EA diagrams from EA tools.

This paper contributes VR-PM, an immersive solution concept towards visualizing and interacting with process models as BPMN [9] and Petri Nets [10][11], process variants, process conformance checking, and PM data in VR. Our prototype realization shows its feasibility, and a case-based evaluation provides insights into its capabilities for addressing the aforementioned challenges. Relative to our initial VR-ProcessMine [1], it offers enhanced capabilities by integrating the visualization of process event logs, Petri Nets, BPMN models, process conformance checking, an alternative process variant visualization, and process discovery, which can generate either a Petri Net or a BPMN model.

This paper is structured as follows: Section II discusses related work. In Section III, the solution is described. Section IV provides details about the realization. The evaluation is described in Section V and is followed by a conclusion.

II. Related Work

The systematic literature review by Milani et al. [12] develops a PM practitioner framework for capturing business questions that PM use cases can answer. While VR and visualization are not mentioned, model discovery (with BPMN and Petri Nets cited as examples) and conformance checking are. Vom Brocke et al. [13] propose a PM research framework that also defines process participants, process stakeholders, and external partners, who are affected by and have an interest in process analyses that span technical, individual, group, organizational, and ecosystem levels. Indeed, our VR-PM solution can support the various roles involved in processes and PM, including behavioral visibility, discovery, variant analysis, and conformance, and enhanced integration with increasing datafication to comprehend and utilize the available information.

As to work regarding PM visualization, the systematic literature review on visualization in PM by Eckhard et al. [14] cites no work involving VR nor any BPMN visualization in conjunction with PM. It explicitly mentions that BPMN and Petri Net PM visualization are under-represented, and conformance-checking PM visualization is under-researched. They conclude that Directly Follows Graph (DFG) is the most commonly used PM visualization. 3DCR [15] is a 3D immersive tool that uses an industrial declarative process modeling language Dynamic Condition Response Graphs (DCR graphs), offering activity-based 3D process animations in the game view using domain-specific representations (sickness registration process). Work involving PM with VR include Vogel and Thomas [16], which shows groundwork and an architecture concept, yet no prototype is described nor are VR screenshots provided. Corea and Delfmann [17] describe a 3D tool for geospatial visualization of paths based on a DFG generated from the PM tool PM4Py. Other work combining PM with VR is often specialized to processes in certain sectors, such as training for factory or manufacturing, logistics, safety, or education and learning, or the health sector. For instance, Roldán et al. [18] describe a complex assembly training system for Industry 4.0 operators.

Open source PM tools include the ProM Framework [19], Apromore [20], and PM4Py [21]; commercial options include products from over 35 vendors, including Celonis, Disco, UiPath, ARIS, and PAFnow. These tools typically provide a 2D user interface with some being Web-based interfaces (e.g., Celonis, UiPath, Apromore), whereas our contribution offers a VR-based solution that can be more engaging and accessible for collaboration with non-experts.

III. Solution

Our solution for PM visualization incorporates VR. VR provides an unlimited space for visualizing a complex set of events, models, and processes and their interrelationships simultaneously in a spatial structure. This provides an immersive, intuitive experience for digital process model visualization and analysis in a 3D space viewable from different perspectives that supports sharing and collaboration among stakeholders. As to benefits of an immersive VR experience vs. 2D for an analysis task, [22] investigated a software analysis task that used a Famix metamodel of Apache Tomcat source code dependencies in a force-directed graph. They found that VR does not significantly decrease comprehension and analysis time nor significantly improve correctness (although fewer errors were made). While interaction time was less efficient, VR improved the UX (user experience), being more motivating, less demanding, more inventive/innovative, and more clearly structured.

To provide context for our solution, our generalized solution concept for VR-PM is shown in Figure 1. VR-PM extends VR-ProcessMine capabilities and leverages our generalized VR Modeling Framework (VR-MF) [7] solution concept. It consists of a VR-based domain-independent hypermodeling framework, which addresses four primary aspects that require special attention when modeling in VR: visualization, navigation, interaction, and data retrieval. Our prior solutions in the EA and BP space, VR-EA [7] provides specialized direct support and mapping for EA models in VR, including both ArchiMate as well as BPMN via VR-BPMN [6]. VR-EAT [8] extends this further with integration of EA tools for accessing dynamically generated diagrams and models from an EA tool in VR. VR-EA+TCK extends these capabilities by integrating further enterprise knowledge, information, and content repositories such as a Knowledge Management System (KMS) and/or an Enterprise Content Management System (ECMS). Our other VR solutions address Software Engineering (SE) and Systems Engineering (SysE) areas.

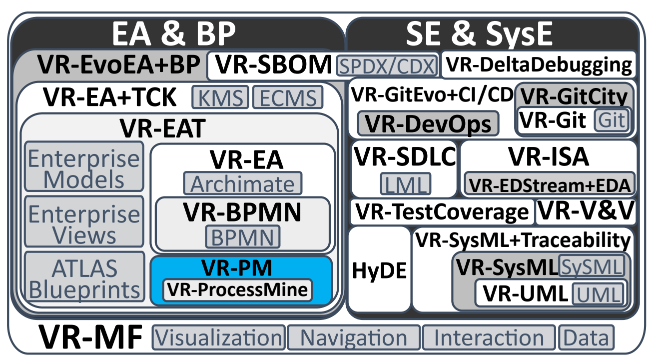

Figure 1. The VR-PM solution concept (blue) in relation to our prior VR solution concepts.

In order to support PM and visual analysis, the VR-PM solution should exhibit the following capabilities:

- *Log file import*: event logs in different event log formats can be imported and processed;

- *Multiple simultaneous analyses*: different event log types (versions or models) can be loaded in order to compare them directly;
- *3D visualization*: elements should be depicted in 3D to support an immersive observation experience;
- *Free element placement*: an individual analysis should be movable in VR space so that they can be compared in locality with another analysis of interest;
- *Hide/show analyses*: to minimize visual clutter, analyses can be hidden and then seen again;
- *Variant depiction*: variants can be analyzed separately as Directly-Follows Graph (DFG) depictions;
- *Colored hot spots*: events are colored to indicate their relative frequency;
- *Log event model*: for reference, show the events before their transformation to other models; and
- *Both formal and practical process models:* offer Petri Nets as a highly formal and rigorous process model for in-depth analysis, as well as a BPMN model that can be practically used and readily understood by novice stakeholders.

A. *Visualization in VR*

In order to differentiate process variants (depending on the analysis being done), our visualization concept for VR depicts each of these on separate vertical plates standing on a common hyperplane representing a single process. This permits any plate to be selected, moved, and compared with others of interest. Furthermore, since the number of process variants can be very large, it leverages the unlimited space in VR, allowing the hyperplane to depict many process variants at once. All process variants are initially equally spaced on the hyperplane and can be compared with each other.

B. *Navigation in VR*

The immersion afforded by VR requires addressing how to navigate the space while reducing the likelihood of potential VR sickness symptoms. Thus, two navigation modes are included in the solution: the default uses gliding controls, enabling users to fly through the VR space and view objects from any angle they wish. Alternatively, teleporting permits a user to select a destination and be instantly placed there (i.e., by instantly moving the camera to that position); while this can be disconcerting, it may reduce the susceptibility to VR sickness for those prone to it that can occur when moving through a virtual space.

C. *Interaction in VR*

Since interaction with VR elements has not yet become standardized or intuitive, in our VR concept, user-element interaction is handled primarily via the VR controllers and a virtual tablet. Our VR-Tablet provides detailed context-specific element information, and can provide a virtual keyboard for text entry fields (via laser pointer key selection) when needed. An affordance in the form of an anchor (sphere) is provided on a corner of any hyperplane or model, which if selected can be used to reduce visual clutter by collapsing (hide) or expanding (show) that object, or the anchor can be used to place the object elsewhere.

IV. Realization

Our solution prototype is partitioned into the Data Hub, our backend for PM and data processing, and the Unity front end responsible for VR visualization (see Figure 2). The Data Hub, based on Python, prepares datasets for visualization. The python library pm4py (Process Mining for Python) [21] is used to convert the imported log files into data objects and data frames. REST APIs are used to access the backend pm4py functionality from the frontend Unity VR environment, which is implemented in C#.

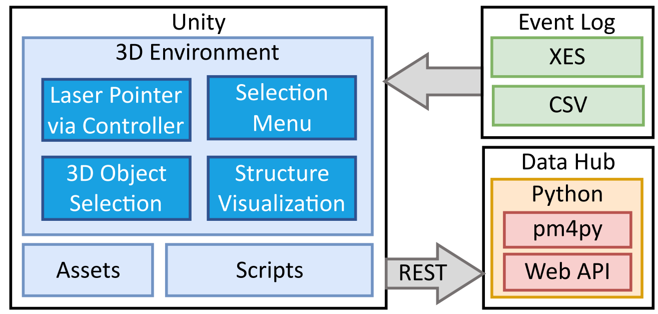

Figure 2. VR-PM logical architecture.

Via our VR-Tablet concept, various configuration options and inputs are supported, as shown in Figure 3.

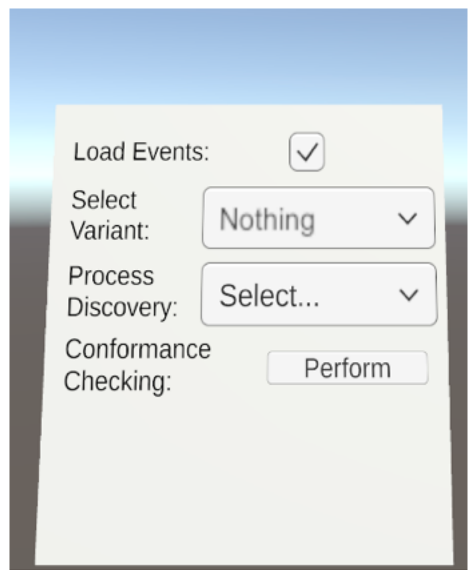

Figure 3. VR-Tablet showing event option, variant selection, process discovery, and conformance checking options.

A. *Event Logs*

An event log is a hierarchical collection of traces. Each trace is a sequence of events that belong to a single case. An event log is required to extract any process variants, which vary in the process paths taken or the activity order. eXtensible Event Stream (XES) [23] or CSV event formats are supported. If no event log exists, but a process model as a BPMN or Petri Net, then via simulation pm4py can generate an event log of various executions.

Apromore [24] was also used to generate and test CSV file support, whereby the simulation offered the ability to specify probability selections pro gateway and included cost.

B. Process Discovery

If an event log exists but no process model, then a process model (Petri Net and BPMN) can be generated via process discovery using PM4Py. However, if a process model such as BPMN exists and no event log, then an event log can be created by process simulation by selecting a Petri Net and then loading the event log in the VR-Tablet, then implicitly a simulation is run based on the model. If an event log already exists, then it is not overwritten. By default, 100 process executions are simulated using PM4Py to generate an event log.

Given an event log, process discovery using PM4Py can generate a Petri Net or a BPMN model based on the data, invoked as shown in Listing 1. Petri Net generation uses the heuristic miner algorithm, while BPMN uses the pm4py inductive miner algorithm. If only a BPMN model is provided as input, then it is converted to a Petri Net. Further detail on the algorithms is provided in the PM4Py documentation.

```
import pm4py

petri_net = pm4py.discover_petri_net_alpha(event_log)
bpmn = pm4py.discover_bpmn_inductive(event_log)
```

Listing 1. Code snippet showing process discovery impementation.

C. Process Conformance Checking

Given a process model as a Petri Net and an event log, pm4py's process conformance via alignment is utilized, as shown in Listing 2. A replay fitness (i.e. alignment) metric score is returned by pm4py, whereby a max value of 1 indicates the behavior contained in the event log is fully conformant to the process model, whereas 0 indicates no conformance whatsoever, and anything in-between partial conformance. Further details are provided in the pm4py documentation.

```
from pm4py.algo.conformance.alignments.decomposed import
    algorithm
from pm4py.algo.evaluation.replay_fitness.algorithm import
    evaluate, Variants

conf = algorithm.apply(event_log, net, im, fm)

alignment_score = evaluate(conf,
                           variant=Variants.ALIGNMENT_BASED)
```

Listing 2. Code snippet showing process discovery impementation.

D. Process Variant Plates

To support process variant analysis, a DFG algorithm provides a summary of all process event transitions and variants and how often each process variant was executed. Figure 4. shows a DFG-based process map visualization result using pm4py. A node represents an event. A graph consists of a set of transitions between a set of nodes.

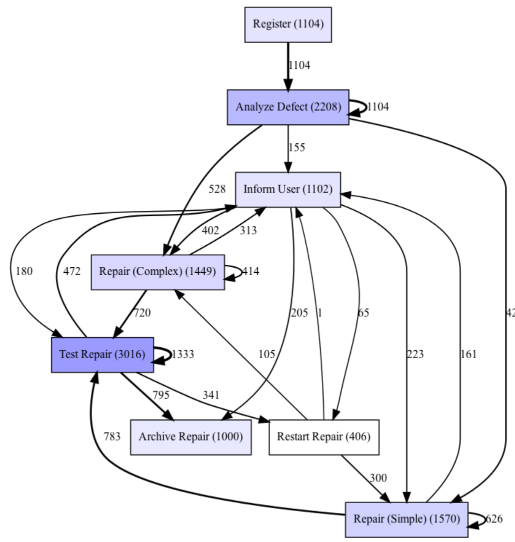

Figure 4. Example DFG-based process map result from pm4py.

Listing 3 shows the result of the parsed dataset providing the number of transitions occurring between two events.

```
Counter({
    ('Test Repair', 'Test Repair'): 1333,
    ('Register', 'Analyze Defect'): 1104,
    ('Analyze Defect', 'Analyze Defect'): 1104,
    ('Test Repair', 'Archive Repair'): 795,
    ('Repair (Simple)', 'Test Repair'): 783,
    ('Repair (Complex)', 'Test Repair'): 720,
    ('Analyze Defect', 'Repair (Complex)'): 528,
    ...
}
```

Listing 3. Snippet of a parsed DFG dataset.

For process variants, data is converted to a dictionary, whereby all duplicates are removed so that each node exists only once in the graph. The recursive list of fan-in relations (nodes reaching this node) together with their occurrence frequency provides the basis for a weighted directed graph as shown in Listing 4. Aggregating the total occurrences across all incoming transitions of a node (event) provides a total frequency of that event across all process instances.

```
{
  "Register": {
    "start":
  },

  "Analyze Defect": {
    "Register": 1104,
    "Analyze Defect": 1104
  },

  "Repair (Complex)": {
    "Analyze Defect": 528,
    "Repair (Complex)": 414,
    "Inform User": 402,
    "Restart Repair": 105
  },
    ...
}
```

Listing 4. Recursive list of fan-in relations.

V. Evaluation

The evaluation of our VR solution concept is based on the design science method and principles [25], in particular a viable artifact, problem relevance, and design evaluation (utility, quality, efficacy). Our case study focuses on addressing the aforementioned challenges identified in the PM Manifesto [5], which are: Improving Usability for Non-experts, Improving Understandability for Non-experts, and Combining Process Mining with other Types of Analysis. The case scenarios are:

- Event Log Visualization and Comprehension
- Process Discovery and Simulation Scenario
- Process Conformance Checking Scenario
- Detailed Process Variant Analysis Scenario
- Combining PM with other Types of Analysis

A. Event Log Visualization and Comprehension

Event logs contain data in formats that are not readily comprehensible for stakeholders. The event log is visually depicted in the form of a 3D nexus rather than text-based data, improving understandability and comprehensibility, as shown in Figure 5. A 3D nexus enables larger graphs to not take up as much space and can be immersively explored without requiring large navigational distances. Events are depicted as spheres (white, green for start, and red for end), directional edges as pipes (darker indicates the source or from side) representing transitions to indicate the sequence direction.

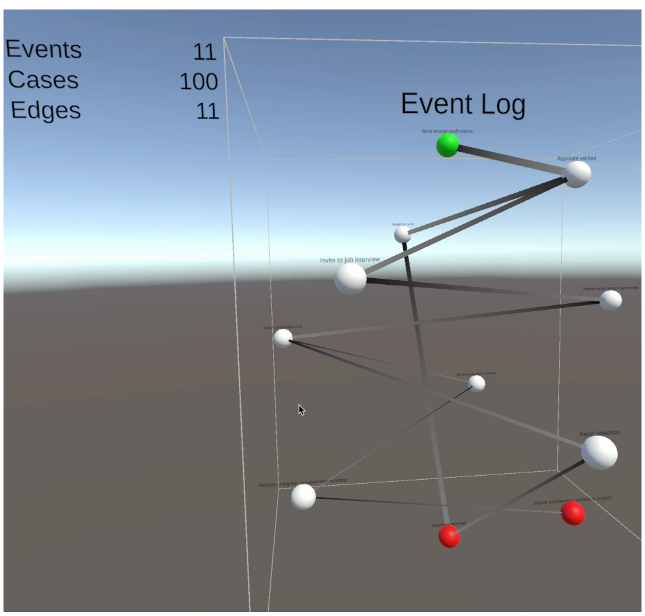

Figure 5. Event log visualized in a nexus form with metrics on top left.

Selecting an event depicts detailed information about it, as shown in Figure 6. Examples of relevant process information include cost information or task duration information, if available in the event log it can be depicted.

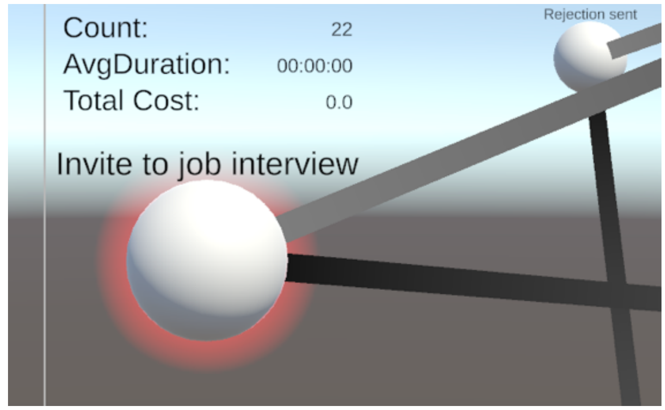

Figure 6. Selected event in event log nexus depicts detailed information related to the event.

B. Process Discovery and Simulation Scenario

Process discovery generates a process model based on event data. To demonstrate process simulation, a BPMN model of a Hiring Process for ACME AG [26] was used, shown in Figure 15. This is depicted in VR-PM in 3D in Figure 16. A Petri Net generated by PM4Py can also be depicted in VR-PM, as shown in Figure 17.

To support basic process variant analysis, the event log can be used to analyze variants within the BPMN model, as shown in Figure 18. Here, activities for any selected variant are highlighted. Only one selected variant is highlighted at a time in BPMN, limiting comparison but supporting simplicity. For more detailed variant analysis, separate variant plate visualization is offered as described later.

To support usability, any process model elements (Petri Net or BPMN) can be repositioned, as shown for the highlighted element in Figure 7 and Figure 8.

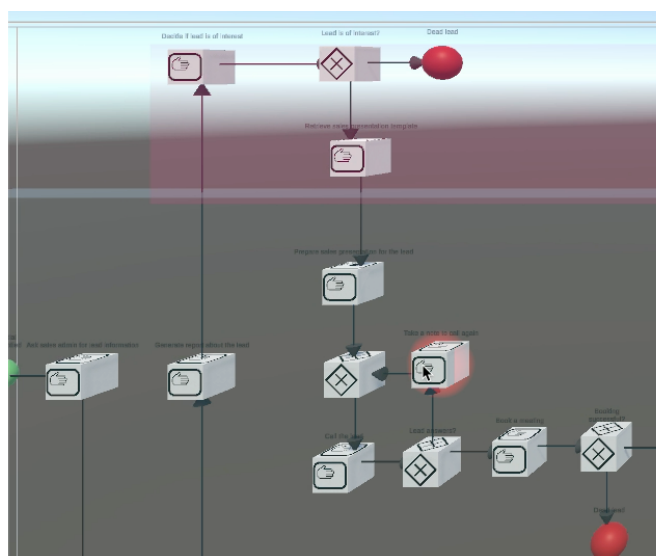

Figure 7. Selection of BPMN process model elements to reposition from default layout.

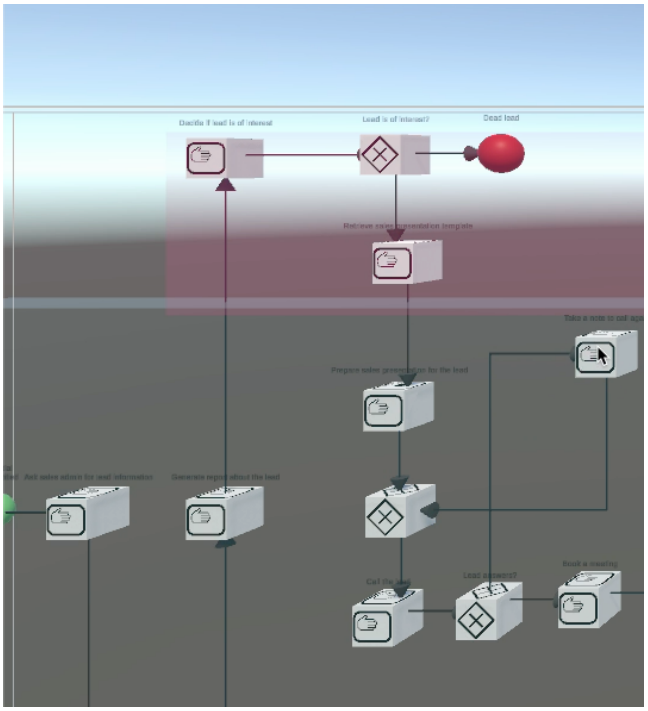

Figure 8. BPMN process model element repositioned.

Process simulation generates traces, which can vary in their event sequences, as shown in Listing 5 versus Listing 6. These result in process variants, each of which contains one or more process instances that followed the same sequence. Selecting a variant in the VR-Tablet result in its path being highlighted in the process model, as was shown in Figure 18. Since the generated BPMN does not necessarily convey all the data available in the event log, a more detailed variant analysis using panes with DFGs is offered, which is described later.

```
<?xml version="1.0" encoding="utf-8" ?>
<log xes.version="1849-2016" xes.features="nested-attributes" xmlns="http://www.
xes-standard.org/">
  <trace>
    <string key="concept:name" value="0" />
    <event>
      <string key="concept:name" value="Ask sales admin for lead information" />
      <date key="time:timestamp" value="1970-04-26T18:46:40+00:00" />
      <string key="case:concept:name" value="0" />
    </event>
    <event>
      <string key="concept:name" value="Check company's financial statements" />
      <date key="time:timestamp" value="1970-04-26T18:46:41+00:00" />
      <string key="case:concept:name" value="0" />
    </event>
    <event>
      <string key="concept:name" value="Check company's web site" />
      <date key="time:timestamp" value="1970-04-26T18:46:42+00:00" />
      <string key="case:concept:name" value="0" />
    </event>
    <event>
      <string key="concept:name" value="Generate report about the lead" />
      <date key="time:timestamp" value="1970-04-26T18:46:43+00:00" />
      <string key="case:concept:name" value="0" />
    </event>
    <event>
      <string key="concept:name" value="Decide if lead is of interest" />
      <date key="time:timestamp" value="1970-04-26T18:46:44+00:00" />
      <string key="case:concept:name" value="0" />
    </event>
  </trace>
```

Listing 5. Example trace snippet from a generated XES event log.

```
<trace>
  <string key="concept:name" value="52" />
  <event>
    <string key="concept:name" value="Ask sales admin for lead information" />
    <date key="time:timestamp" value="1970-04-26T18:54:40+00:00" />
    <string key="case:concept:name" value="52" />
  </event>
  <event>
    <string key="concept:name" value="Check company's financial statements" />
    <date key="time:timestamp" value="1970-04-26T18:54:41+00:00" />
    <string key="case:concept:name" value="52" />
  </event>
  <event>
    <string key="concept:name" value="Check company's web site" />
    <date key="time:timestamp" value="1970-04-26T18:54:42+00:00" />
    <string key="case:concept:name" value="52" />
  </event>
  <event>
    <string key="concept:name" value="Generate report about the lead" />
    <date key="time:timestamp" value="1970-04-26T18:54:43+00:00" />
    <string key="case:concept:name" value="52" />
  </event>
  <event>
    <string key="concept:name" value="Decide if lead is of interest" />
    <date key="time:timestamp" value="1970-04-26T18:54:44+00:00" />
    <string key="case:concept:name" value="52" />
  </event>
  <event>
    <string key="concept:name" value="Retrieve sales presentation template" />
    <date key="time:timestamp" value="1970-04-26T18:54:45+00:00" />
    <string key="case:concept:name" value="52" />
  </event>
  <event>
    <string key="concept:name" value="Prepare sales presentation for the lead" />
    <date key="time:timestamp" value="1970-04-26T18:54:46+00:00" />
    <string key="case:concept:name" value="52" />
  </event>
  <event>
    <string key="concept:name" value="Call the lead" />
    <date key="time:timestamp" value="1970-04-26T18:54:47+00:00" />
    <string key="case:concept:name" value="52" />
  </event>
  <event>
    <string key="concept:name" value="Book a meeting" />
    <date key="time:timestamp" value="1970-04-26T18:54:48+00:00" />
    <string key="case:concept:name" value="52" />
  </event>
  <event>
    <string key="concept:name" value="Do the meeting" />
    <date key="time:timestamp" value="1970-04-26T18:54:49+00:00" />
    <string key="case:concept:name" value="52" />
  </event>
</trace>
```

Listing 6. Further trace snippet sequence from a generated XES event log.

To demonstrate scalability and immersion in VR, a larger BPMN insurance company model [25], containing 27 activities and 13 gateways, was evaluated, shown in Figure 19. This was loaded into VR-PM as shown in Figure 20.

C. *Process Conformance Checking Scenario*

Process conformance checking can be initiated by the button on the VR-Tablet, as shown in Figure 21. The results are displayed at the top of the VR-Tablet, showing the percentage of fitting traces, the average fitness, and the log fitness. This figure also shows multiple model representations (event log nexus, BPMN, and Petri Net) simultaneously, which may be helpful when analyzing conformance issues.

D. *Detailed Process Variant Analysis Scenario*

For detailed DFG-based process variant analysis, the dataset consisted of randomly generated process variants based on a software defect repair process (a snippet is shown in Listing 7). A process variant represents multiple process instances, whereby any process instance exhibits the same sequence (or node transition) order.

```
{
"variant": "Register,Analyze Defect,Analyze Defect,Inform User,
Repair (Complex),Repair (Complex),Test Repair,Test Repair,Archive
Repair",
"case:concept:name": 78
},
{
"variant": "Register,Analyze Defect,Analyze Defect,Repair (Simple),
Inform User,Repair (Simple),Test Repair,Test Repair,Archive
Repair",
"case:concept:name": 75
},
{
"variant": "Register,Analyze Defect,Analyze Defect,Repair (Simple),
Repair (Simple),Test Repair,Test Repair,Inform User,Archive
Repair",
"case:concept:name": 67
},
{
"variant": "Register,Analyze Defect,Analyze Defect,Repair
(Complex),Repair (Complex),Test Repair,Inform User,Test Repair,
Archive Repair",
"case:concept:name": 64
},
{
"variant": "Register,Analyze Defect,Analyze Defect,Repair (Simple),
Repair (Simple),Test Repair,Inform User,Test Repair,Archive
Repair",
"case:concept:name": 41
},
{
"variant": "Register,Analyze Defect,Analyze Defect,Repair (Simple),
Repair (Simple),Test Repair,Test Repair,Restart Repair,Repair
(Simple),Inform User,Repair (Simple),Test Repair,Test Repair,
Archive Repair",
"case:concept:name": 29
},
{
"variant": "Register,Analyze Defect,Analyze Defect,Inform User,
Repair (Simple),Repair (Simple),Test Repair,Test Repair,Archive
Repair",
"case:concept:name": 28
},
{
"variant": "Register,Analyze Defect,Analyze Defect,Repair (Simple),
Repair (Simple),Test Repair,Test Repair,Inform User,Restart Repair,
Repair (Simple),Repair (Simple),Test Repair,Test Repair,Archive
Repair",
"case:concept:name": 21
},
```

Listing 7. Dataset snippet of randomized process variants (in JSON).

For variant analysis, the initial variant plate shows a DFG with entire set of nodes and transition frequency; the plates behind it depict the different process variants, as shown in Figure 9. Each process variant is rendered on a separate 2D plate stacked horizontally (as depth), thus the third dimension allows the user to readily perceive the number of process variants.

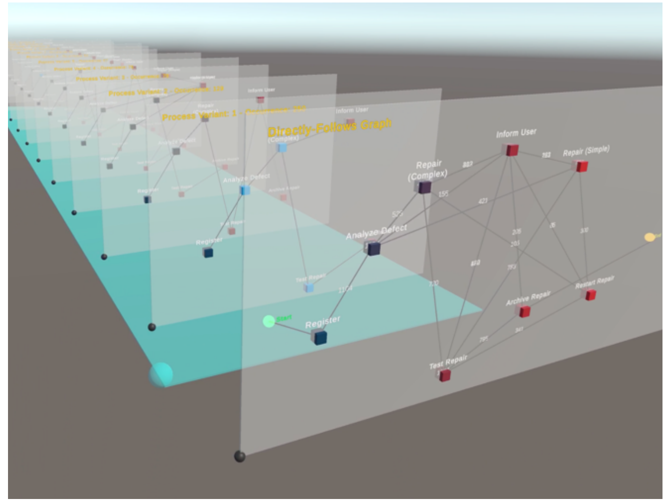

Figure 9. Initial plate shows DFG with entire set of nodes and transition frequency [1]. Each plate thereafter represents one process variant.

The variants are assigned a default ID with their occurrence frequency indicated on top, as seen in Figure 10.

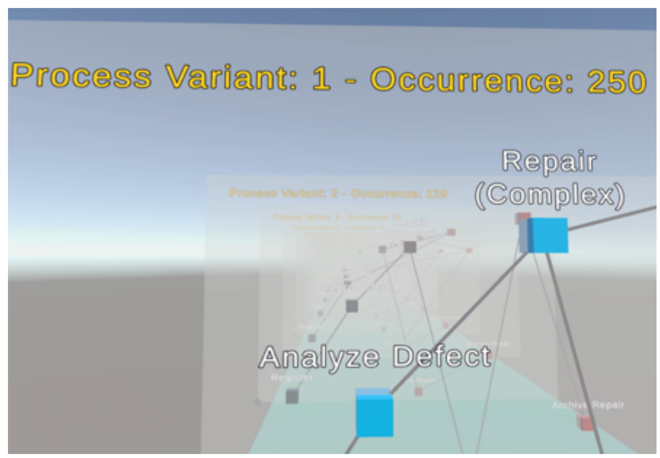

Figure 10. Partial process variant showing occurence frequency [1].

The total number of input transitions to a node represent the total number of times that event occurred. Thus, the higher this number, the higher the frequency. To represent this visually, a ten-step color scale was used to map the frequency between low activity (blue) and high activity (red), analogous to mapping temperature, as shown in Figure 11. This can be used to quickly identify frequently occurring events in a process and help focus analysis and potential optimizations.

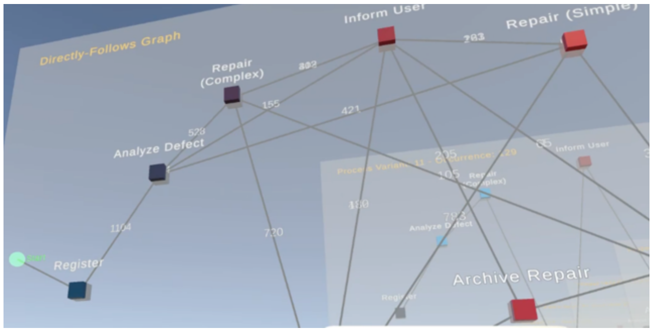

Figure 11. Node color and edges in a DFG, from VR-ProcessMine [1].

As plates on a hyperplane, the relation to the process model is supported by the hyperplane, yet each variant can be considered separately and compared with others by moving them with the affordance in the bottom left corner, as shown in Figure 12. One can also collapse any that are irrelevant to the analysis to reduce visual clutter.

A large set of different process models, and their variants, can be depicted simultaneously, supporting larger cross-process analysis scenarios as seen in Figure 13.

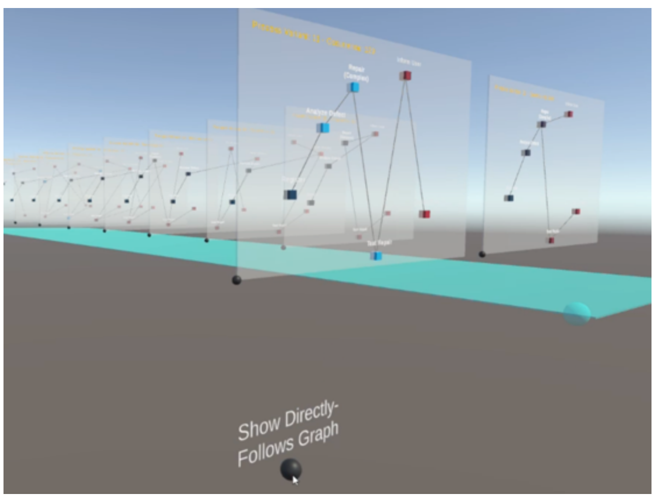

Figure 12. Anchor control for variant comparison or collapsing/expanding [1].

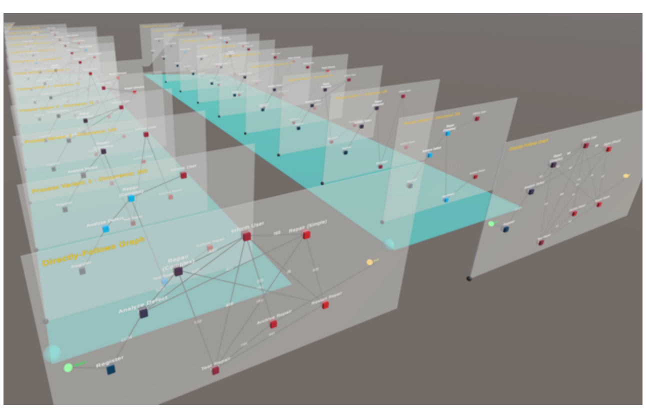

Figure 13. Different processes can be displayed via different hyperplanes [1].

E. Combining PM with other Types of Analysis

PM analysis can provide significant operational insights, yet additional enterprise knowledge or data may also be relevant. Towards addressing this challenge, one advantage of VR's unlimited space is its ability to represent multiple heterogenous models simultaneously (Figure 21), which non-experts can immersively explore. Further heterogeneous hypermodeling capabilities in VR are exemplified with our VR-EAT [8], which is based on diagrams generated by Atlas and transformed in VR, and with our VR-EA [7], offering VR-based ArchiMate diagrams, as shown in Figure 14. Thus, VR-PM models can be placed side-by-side with other enterprise models in VR, making deeper cross-model and cross-domain analysis readily accessible to stakeholders, while enabling operational insights to guide such analyses.

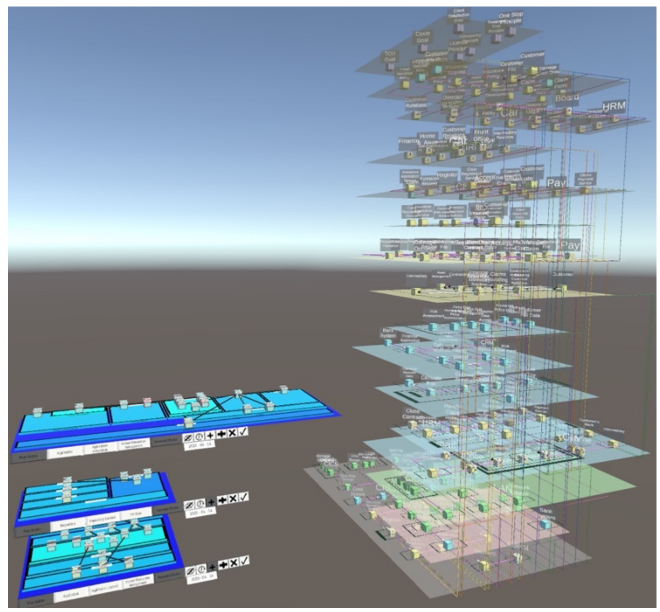

Figure 14. VR-EAT [8] enterprise hypermodeling and analysis: Atlas-based diagrams (left) and VR-EA [7] using ArchiMate-based diagrams (right).

F. Discussion

Towards addressing the aforementioned three PM challenges, our solution concept seeks to improve the usability and understandability for non-experts and enabling the combination with other analyses. It does this by providing an immersive VR experience addressing visualization, navigation, interaction, and data retrieval. This was shown via common PM case scenarios:

- Event Log Visualization and Comprehension
- Process Discovery and Simulation Scenario
- Process Conformance Checking Scenario
- Detailed Process Variant Analysis Scenario
- Combining PM with other Types of Analysis

On the basis of these PM support scenarios, the solution can further PM stakeholder engagement and collaboration, such that they can intuitively interact with and comprehend process models and evidence-based process event data, and readily incorporate these in their daily work routines.

VI. Conclusion

The increasing digitalization in enterprises and organizations implies that the business, operational, and industrial processes executed will increasingly also become digitally accessible, offering a significant opportunity for further uptake and engagement of stakeholders, such as enterprise citizens. While current PM tools and techniques can provide valuable insights for optimizing (business) processes, these benefits can be hindered if insights are not readily accessible to a larger (non-expert) stakeholder set, such as those directly involved in performing these processes.

VR-PM contributes an immersive solution concept for visualizing and interacting with process models (BPMN and Petri Nets), process variants, process conformance checking, and PM data in VR. Our realization shows its feasibility, and the case-based evaluation provides insights into its capabilities towards addressing certain challenges described in the Process Mining Manifesto, in particular improving usability, understandability, and the potential to combine PM with other types of analysis.

Future work includes offering native editing capabilities to the process models, the depiction of message flows, more comprehensive PM analyses, deeper integration with our enterprise hypermodeling VR-EA-TCK and VR-BPMN solution concept, automatic filtering of process variants by a node or transition of interest, enhanced collaboration support, and a comprehensive empirical study.

Acknowledgment

The authors would like to thank Maurizio Jenak, Laura Berroth, and Michael Scholl for their assistance with the design, implementation, figures, and evaluation.

References

[1] R. Oberhauser, “VR-ProcessMine: Immersive Process Mining Visualization and Analysis in Virtual Reality,” In: Proceedings of the Fourteenth International Conference on Information, Process, and Knowledge Management (eKNOW 2022), IARIA, 2022, pp. 75-80.

[2] Grand View Research, Inc., “Business Process Management Market (2025 - 2030).” [Online]. Available from: https://www.grandviewresearch.com/industry-analysis/business-process-management-bpm-market 2025.11.28

[3] W. M. van der Aalst, “Process mining: a 360 degree overview,” In: Process mining handbook, Cham: Springer International Publishing, 2022, pp. 3-34.

[4] W. M. van der Aalst. Process Mining - Data Science in Action, Second Edition. Springer, 2016.

[5] W. V. D. Aalst, et al., "Process Mining Manifesto," In: International Conference on Business Process Management. Springer, Berlin, Heidelberg, 2011, pp. 169-194.

[6] R. Oberhauser, C. Pogolski, and A. Matic, "VR-BPMN: Visualizing BPMN models in Virtual Reality," In: Shishkov, B. (ed.) BMSD 2018. LNBIP, vol. 319, Springer, Cham, 2018, pp. 83–97. https://doi.org/10.1007/978-3-319-94214-8_6

[7] R. Oberhauser and C. Pogolski, "VR-EA: Virtual Reality Visualization of Enterprise Architecture Models with ArchiMate and BPMN," In: Shishkov, B. (ed.) BMSD 2019. LNBIP, vol. 356, Springer, Cham, 2019, pp. 170–187. https://doi.org/10.1007/978-3-030-24854-3_11

[8] R. Oberhauser, P. Sousa, and F. Michel, "VR-EAT: Visualization of Enterprise Architecture Tool Diagrams in Virtual Reality," In: Business Modeling and Software Design. BMSD 2020. LNBIP, vol 391, Springer, Cham, 2020, pp. 221-239. https://doi.org/10.1007/978-3-030-52306-0_14

[9] OMG, "Business Process Model and Notation (BPMN) Version 2.0.2," 2014.

[10] J.L. Peterson, "Petri nets," ACM Computing Surveys (CSUR), 9(3), 1977, pp. 223-252.

[11] W. Reisig. Understanding Petri Nets. Springer-Verlag Berlin, 2016. doi: 10.1007/978-3-642-33278-4.

[12] F. Milani, K. Lashkevich, F.M. Maggi, and C. Di Francescomarino, “Process mining: a guide for practitioners,” In: International conference on research challenges in information science, Springer, Cham, 2022, pp. 265-282.

[13] J. Vom Brocke, M. Jans, J. Mendling, and H.A. Reijers, „A five-level framework for research on process mining,” Business & Information Systems Engineering, 63(5), pp. 483-490, 2021.

[14] F. Eckhard, H. Wittges, and S. Rinderle-Ma, “Exploring the Role of Visualization in Process Mining: A Systematic Literature Review.” In: International Conference on Business Information Systems, Springer, Cham, 2025, pp. 165-178.

[15] J.C. Villalobos, S.J. Jensen, and H.A López-Acosta, “3DCR: A Tool for Immersive Process Mining,” In: 6th International Conference on Process Mining. CEUR-WS, 2024.

[16] J. Vogel and O. Thomas, "Towards a virtual reality-based process elicitation system," 40 Years EMISA 2019. LNI P-304, Gesellschaft für Informatik e.V., 2020, pp. 91-104, ISBN 978-3-88579-698-5

[17] C. Corea and P. Delfmann, “Geo-aware process mining,” In: Proc.Best Dissertation Award, Doctoral Consortium, and Demo. & Resources Forum at BPM, 2024, pp. 101-105.

[18] J. J. Roldán, E. Crespo, A. Martín-Barrio, E. Peña-Tapia, and A. Barrientos, "A training system for Industry 4.0 operators in complex assemblies based on virtual reality and process mining," Robotics and computer-integrated manufacturing, 59, 2019, pp. 305-316.

[19] B. F. Van Dongen, A. K. A. de Medeiros, H. Verbeek, A. Weijters, and W. van der Aalst, “The ProM Framework: A new era in Process Mining Tool Support,” In: International Conference on Application and Theory of Petri Nets, Springer, 2005, pp. 444–454.

[20] M. La Rosa et al., "APROMORE: An advanced process model repository," Expert Systems with Applications, 38(6), 2011, pp. 7029-7040.

[21] A. Berti, S. van Zelst, and D. Schuster, "PM4Py: A process mining library for Python," Software Impacts, 17, p. 100556, 2023. doi: 10.1016/j.simpa.2023.100556

[22] R. Müller, P. Kovacs, J. Schilbach, and D. Zeckzer, "How to master challenges in experimental evaluation of 2D versus 3D software visualizations," In: 2014 IEEE VIS International Workshop on 3Dvis (3Dvis), IEEE, 2014, pp. 33-36.

[23] IEEE, "IEEE Standard for eXtensible Event Stream (XES) for Achieving Interoperability in Event Logs and Event Streams," in IEEE Std 1849-2023, 2023.

[24] Apromore. [Online]. Available from https://academic-cloud.apromore.org 2025.11.28

[25] A.R. Hevner, S.T. March, J. Park, and S. Ram, "Design science in information systems research," MIS Quarterly, 28(1), 2004, pp. 75-105.

[26] D. Sola, C. Warmuth, B. Schäfer, P. Badakhshan, J.-R. Rehse, and T. Kampik, “SAP Signavio Academic Models: A Large Process Model Dataset,” in Process Mining Workshops - ICPM 2022 International Workshops, October 23-28, 2022, Revised Selected Papers, 2022, vol. 468, pp. 453–465. doi: 10.1007/978-3-031-27815-0_33.

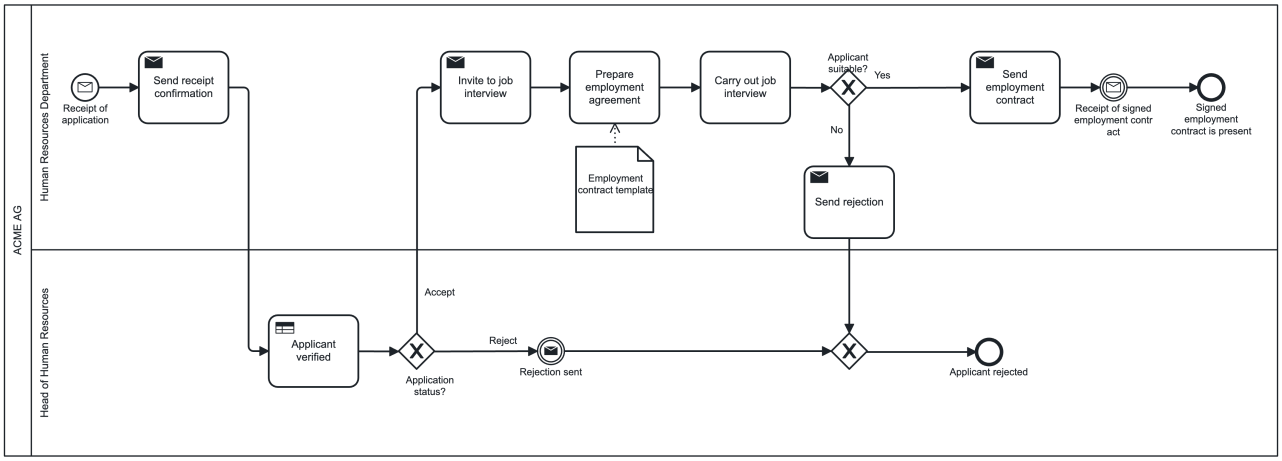

Figure 15. A BPMN model diagram of a hiring process for ACME AG [26].

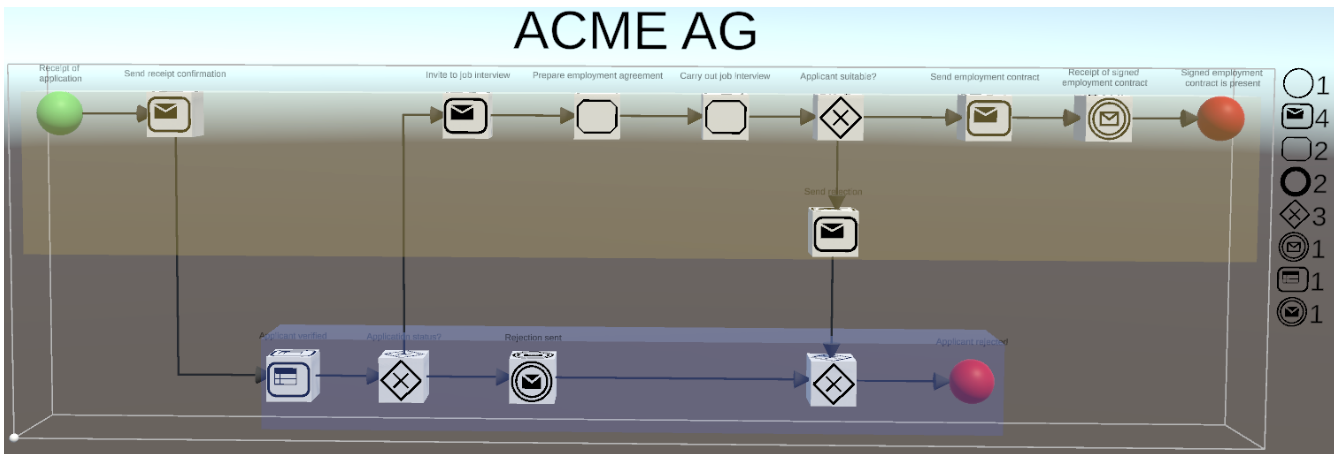

Figure 16. A BPMN model (ACME AG) depicted in 3D in VR-PM, showing count metrics on the legend on the right.

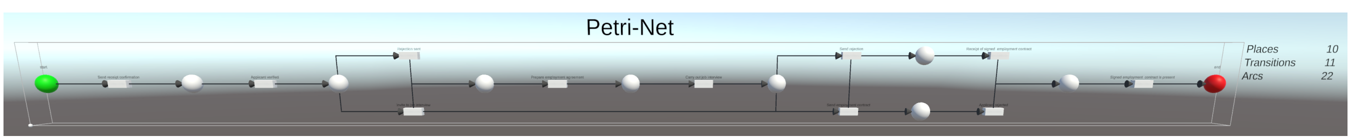

Figure 17. Petri Net 3D visualization in VR-PM.

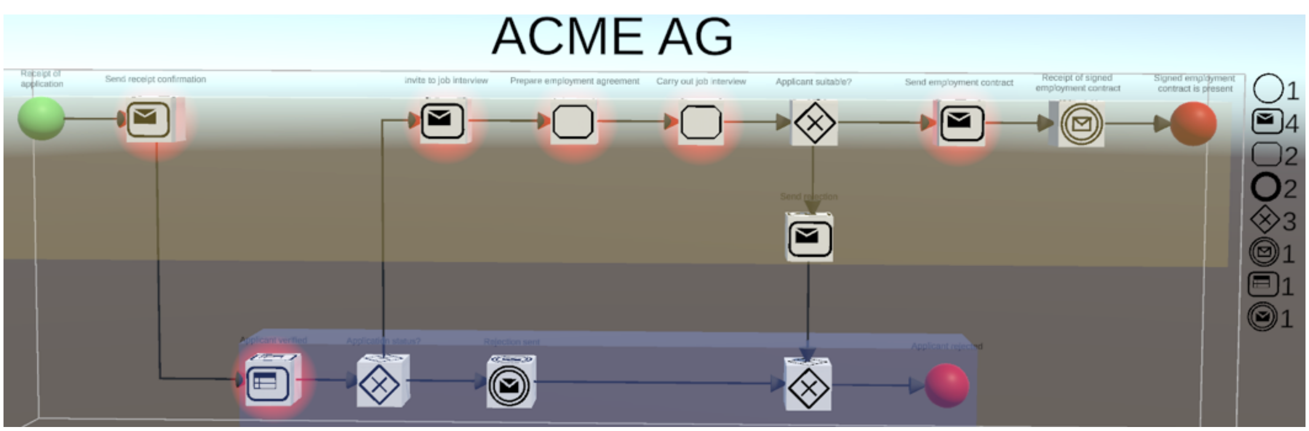

Figure 18. VR-PM variant selection within the BPMN model highlights its corresponding activities.

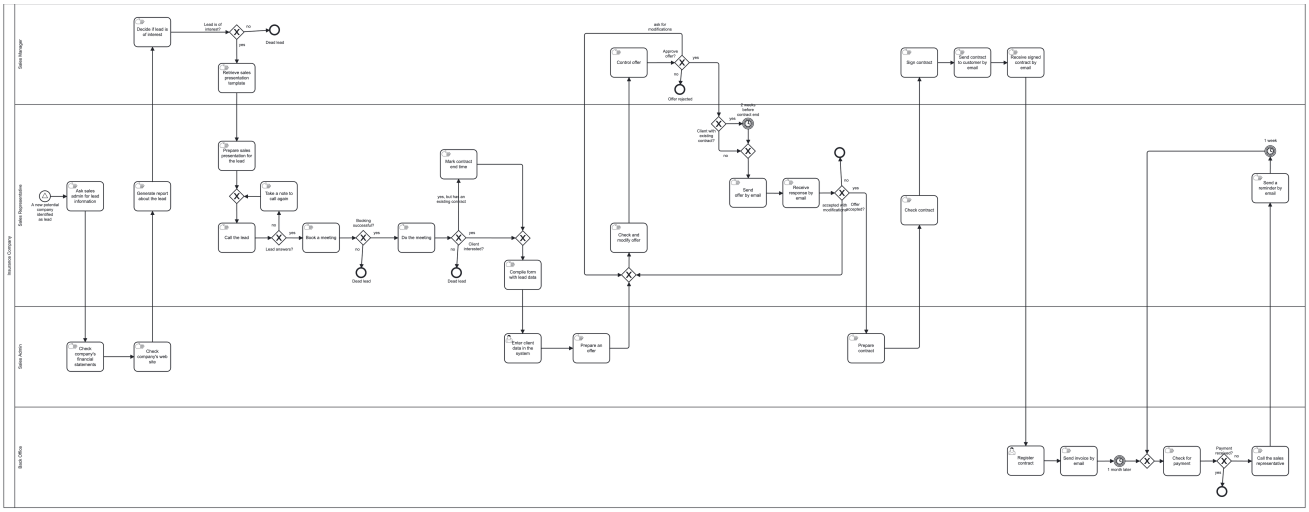

Figure 19. Large BPMN model diagram (Insurance Company) [26].

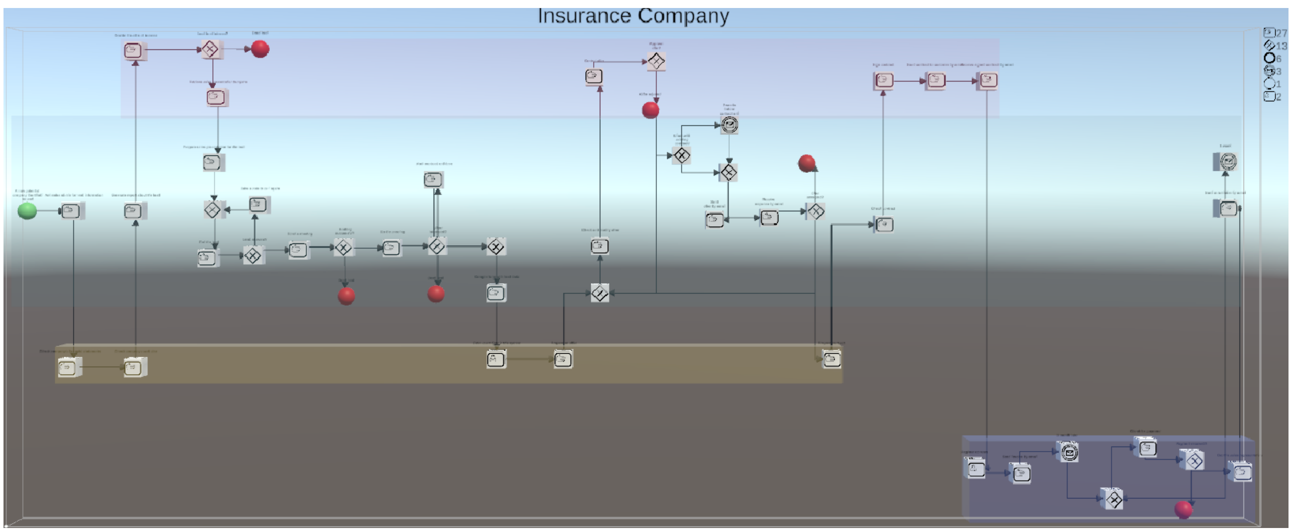

Figure 20. The large BPMN model (Insurance Company) depicted in 3D in VR-PM.

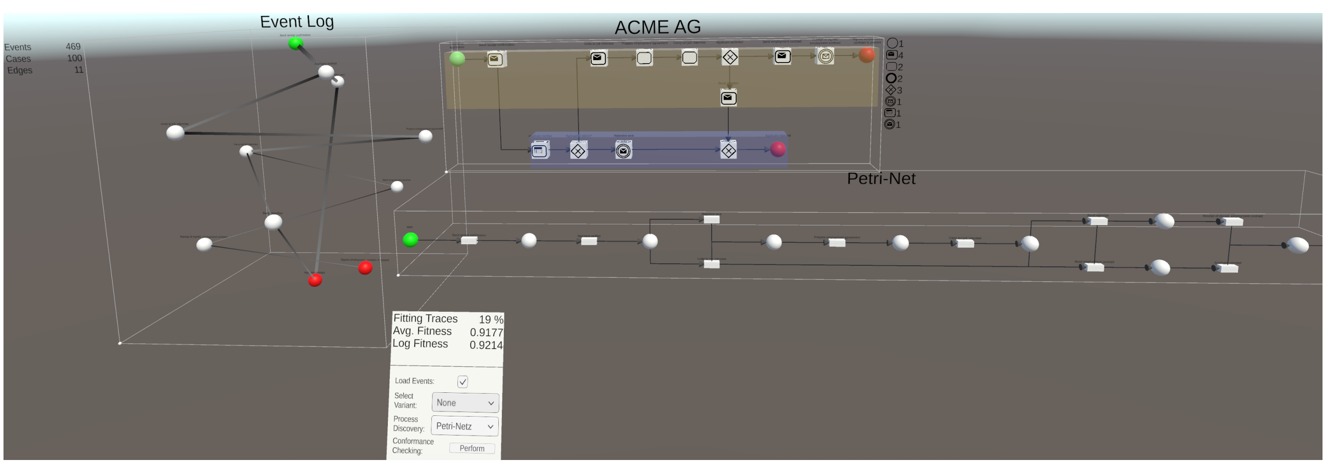

Figure 21. VR-PM showing multiple model representations simultaneously (event log nexus left, BPMN top, Petri Net bottom) with compliance metrics.