VESPA: Visual Event-Stream Progressive Analytics

Andrea Burattin[1] [0000−0002−0837−0183], Silvia Miksch[2] [0000−0003−4427−5703], Shazia Sadiq[3] [0000−0001−6739−41455], Hans-Jörg Schulz[4] [0000−0001−9974−535X], and Katerina Vrotsou[5] [0000−0003−4761−8601]

[1] Technical University of Denmark, Kgs. Lyngby, Denmark, andbur@dtu.dk
[2] TU Wien, Vienna, Austria, silvia.miksch@tuwien.ac.at
[3] The University of Queensland, Australia, shazia@eecs.uq.edu.au
[4] Aarhus University, Aarhus, Denmark, hjschulz@cs.au.dk
[5] Linköping University, Campus Norrköping, Sweden, katerina.vrotsou@liu.se

Abstract. This paper introduces VESPA (Visual Event-Stream Progressive Analytics), a framework that integrates Streaming Process Mining (SPM) with Progressive Visual Analytics (PVA) to support timely, informed decisions using partial, evolving data. VESPA addresses the challenges of analyzing transient, multifaceted event-streams by coupling process mining with progressive visualizations. Our framework is structured around key dimensions (context, task, data, algorithm, user roles, and interaction modalities). Two central research questions guide our work: identifying optimal timing for progressive visualizations and determining their effectiveness and appropriateness in streaming contexts. A prototype with ward-centric and patient-centric views was conceptualized, based on a simulated real-world scenario in the context of an emergency department (ER). These views support user roles from passive monitoring to active exploration, enabling dynamic prioritization and resource allocation. Preliminary results demonstrate the potential of VESPA to enhance situational awareness and decision-making.

Keywords: Streaming Process Mining · Progressive Visual Analytics

1 Introduction

In dynamic, high-stakes environments, such as emergency response (ER) departments, decision-makers must act based on incomplete and continuously evolving data. Traditional process mining techniques, which rely on static, retrospective data, fall short in such contexts. Streaming Process Mining (SPM) addresses this limitation, but often lacks the interpretability needed for decision-making.

This paper proposes the integration of SPM with Progressive Visual Analytics (PVA), which supports sensemaking through incremental visual feedback and interaction, even when data or computations are incomplete. Together, SPM and PVA form a unified framework—VESPA (Visual Event-Stream Progressive Analytics)—that empowers users to monitor, explore, and act on streaming data in real-time. By coupling algorithmic insights with visualization, VESPA supports various user roles and tasks, from passive observation to active exploration.

Consider the real-time decision-making needed when managing a busy ER. Patients undergo a sequence of diagnosis and possibly treatment steps that can be conceptually captured as a process model, which may change depending on the time of day (working hours vs. after hours) and case load (business as usual vs. state of emergency). The head of the ER needs to monitor the current intake, throughput, and related KPIs, like the length of stay (LOS) or ward load (WL), to decide in real-time whether to allocate additional resources (activate on-call doctors), to fast-track certain patients (increase their urgencies), explicitly switch from the usual procedures to the streamlined emergency procedures or back, etc.

To support such time-critical decisions in real-time scenarios, we propose to combine SPM with PVA. Our key contributions include: **(1)** A conceptual framework for integrating SPM and PVA in time-critical decision-making contexts; **(2)** Definition of a multi-dimensional problem space; **(3)** VESPA-VIS: a prototype visualization system showcasing views in a high-stakes real-time environment.

2 Related Work

Streaming Process Mining (SPM). SPM [4] techniques have emerged to handle the processing of constantly updating, real-time information. In a streaming setting, events are processed, immediately after they are generated, by a SPM pipeline, yielding intermediate. SPM algorithms can be used to handle the control-flow discovery, where the control-flow is expected to represent the process *currently* being executed [6]. Another problem that can be tackled is streaming conformance checking [4], where the conformity of each event is verified against a corresponding reference model. Despite these benefits of SPM, in a real-time scenario as described above, the effectiveness of SPM cannot be fully realized without providing effective intermediate insights.

Progressive Visual Analytics (PVA). PVA refers to iterative or incremental approaches for the visual analysis of large amounts of data that could not be processed, visualized, or interacted with in a traditional, whole-dataset-at-once manner [7]. PVA produces partial results, which are usually the outcome of some technical process—e.g., a running computation that refines over time or a complex data query yielding more and more data over time. These intermediate results are then utilized for making time-critical decisions—e.g., in disaster recovery and emergency response scenarios [7, ch.7.6]. Hence, PVA focuses mainly on displaying the progression of intermediate results to allow such decisions, but less on providing insight into the progressive process itself. This is where the combination with SPM comes into the picture.

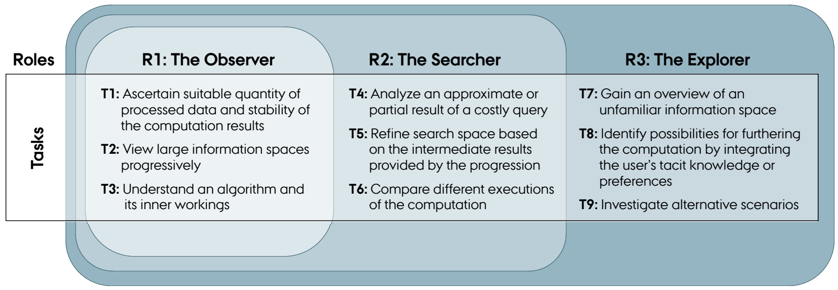

Fig. 1. Common user roles and tasks in PVA. Ranging from observers with limited involvement and interaction possibilities to explorers directly wrangling with one or more running processes in parallel. (adapted from [7, ch.7] under CC-BY 4.0)

3 Our Approach: VESPA

In developing our approach for VESPA, we first outline the problem space, from which we derive two key research questions. We then present the VESPA architecture followed by a demonstration of preliminary results.

Aspects of the problem space. VESPA requires consideration of the following aspects: **(1)** Context, i.e., business process; **(2)** Task, e.g., conformance checking or process enhancement; **(3)** Data space assumes at least an event log but it could be augmented with other facets relevant to the problem; **(4)** Algorithm space, i.e., the specific algorithm relevant to the task; **(5)** Guards/Rules [9] that signal potential attention triggers for users; **(6)** Users include full spectrum from monitoring scenario all the way to a fully explorative scenario following the categorization into Observers, Searchers, Explorers [11] (see Figure 1); **(7)** Visualization and Interaction space, e.g., interactive visualizations of event sequences [2] and dynamic networks [3, 8].

Research questions. We defined two research questions (RQs) posed by such an SPM setting with a continuous flow of events:

RQ1 *What are the required time points for progressive visualization for streaming process mining?* Identification of time points is related to the needs of the analytical intention of the user. Based on previous experiences in related projects, we identified three needs that may arise at different time points and refer to them as scheduled, triggered, and on demand: *(i)* Scheduled (e.g., results are ready); *(ii)* Triggered (e.g., a guard/rule fires when a conformance score is falling below threshold); *(iii)* On demand (e.g., an explorer wants to probe on a particular facet, such as the trend in urgency levels). These time points in turn influence the suitability of the visualization and interaction which leads us to our second research question.

RQ2 *What are the expressive, effective, and appropriate progressive visualizations and interactions for streaming process mining?* *Expressiveness* refers to the requirement of visualizing exactly the information contained in the data; nothing more and nothing less [10]. *Effectiveness* considers the de-

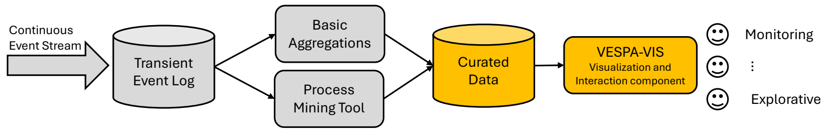

Fig. 2. VESPA's Architecture: The proposed software architecture for PVA in SPM.

gree to which the visualization addresses the cognitive capabilities of the human visual system, the task at hand, the application background, and other context-related information, to obtain intuitively recognizable and interpretable visual representations [10]. Finally, *appropriateness* involves a cost-benefit ratio to assess the value of the visualization process with respect to achieving a given task [15]. For VESPA, this implies that the information displayed should answer to the exact needs, nature and urgency of the current request (expressiveness), they should be made available in a manner that considers these while enabling the user to accomplish their tasks (effectiveness) and retain a cost-benefit balance (appropriateness).

VESPA's Architecture. The overall approach is proposed to be embedded in a software architecture as seen in Figure 2. The continuous event-stream is a key feature of the problem space. Depending on the velocity of the event-stream, there may or may not be a persistent storage and hence the system architecture presents it as a 'transient' event log. A process mining tool is selected based on the task e.g. conformance checking. In addition to the discovered process, the proposed system architecture also produces a process mining results dataset. This includes details such as conformance scores and multi-faceted event data [14]. When needed, the transient event log may also be used to produce basic aggregations such as patient load over a period of time. Together the aggregations and the process mining results constitute a curated dataset that forms the input to the visualization. The results from the visualization component are expected to empower users to perform a range of tasks from monitoring all the way to interactive exploration to support timely (or even real-time) decision making.

Preliminary results. We outline a realistic hypothetical user story expressed in two levels of detail to frame and guide our VESPA approach.

– **Patient-centric:** An ER administrator wants to know if the LOS for one ER patient is too high requiring to prioritize them in the waiting queue.
– **Ward-centric:** An ER administrator wants to know if the LOS for a cohort of ER patients is too high / too low to adjust the allocation of resources.

This user story manifests in the problem dimensions: **(1)** Context: Healthcare; **(2)** Task: Primarily we will focus on conformance checking, but this is intended to be augmented with relevant facets; **(3)** Data space: Ideally, data should come from the information system of the ER; **(4)** Algorithm space: We can use behavioral conformance checking (BCC) [4]; **(5)** Progressive Guards [9]: Three guards are considered: conformance falling below threshold; a load of an urgent category (critical/high) increasing over a threshold; and the LOS of a given patient increasing over a threshold; **(6)** Users: Interchangeable roles of Observers, Searchers, Explorers; and **(7)** Visualization and Interaction space is aligned with the patient-centric and ward-centric user stories. We assume that the processes happening in the ER are not stationary and can change according to different criteria. For example, we can assume that a typical ER process changes during off-peak (with fewer doctors available) or during an intense scenario (with many patients checked at once and doctors coming from other wards).

A SPM pipeline can be implemented using pyBeamline [5]. The pipeline processes each event and computes the following: **(1)** The DFG model [5, 1] updated up to the given point in time; **(2)** The conformance value of the stream against the ER's operational models (off-peak vs intense).

VESPA's Prototypical Mock-Up. Based on the use case and outlined problem dimensions, we designed a prototype mock-up, VESPA-VIS. VESPA-VIS comprises two main views addressing the two levels of detail of the user story, the *Ward view* (cf. Fig. 3) and the *Patient view* (cf. Fig. 4), designed to address the two outlined levels of detail of our ward-centric and patient-centric use cases.

The *Ward view* is split into a *Patient Flow* view (cf. Fig. 3A) and a *Temporal Facet* view (cf. Fig. 3B). The *Patient Flow* displays the event-streams flowing into the ER ward. A node-link diagram representing the currently active reference model is shown as a backdrop in the view. As the events stream into the ward model, the current patient load is mapped to node sizes and edge widths. Deviations to the model (i.e., "unexpected" events not included in the reference model) are drawn in a dashed style. The *Patient Flow* displays the flow of patients over a given expert-user defined time-interval preceding the current time point (e.g., 10 minutes). A time slider allows exploration of past intervals. Hovering over a node or edge pops up a tooltip displaying the number of patients belonging to the corresponding event or transition over time. The *Temporal Facet* view displays a selection of graphs showing the temporal distribution of relevant facets (cf. Fig. 3B). These facet graphs complement the *Patient Flow* view and allow an expert to inspect surrounding factors and reason about the processing state of the ward. On the top, the *conformance score* over time is displayed. Conformance w.r.t. the currently explored model is displayed by default (e.g., regular operations), and on hover, conformance w.r.t. two complementary model variations is also shown (e.g., off-peak, intense). This graph allows the expert to monitor the conformance of the process over time, detect fluctuations from the expected behavior, and assess whether the correct model is used as a reference or whether another model should be used. If the conformance score falls below a certain threshold over a certain period of time, a guard is triggered, calling for the attention of the expert. The second graph displays the *ward load* over time, i.e. the total number of patients being processed, allowing the expert to monitor the overall stress on the ward over time. Third, the average *LOS* of patients being processed is displayed over time, providing an additional cue to the load on the ward. The fourth graph gives a summary overview of the urgency of the patients being processed over time. The distribution of the urgency classes (e.g.,

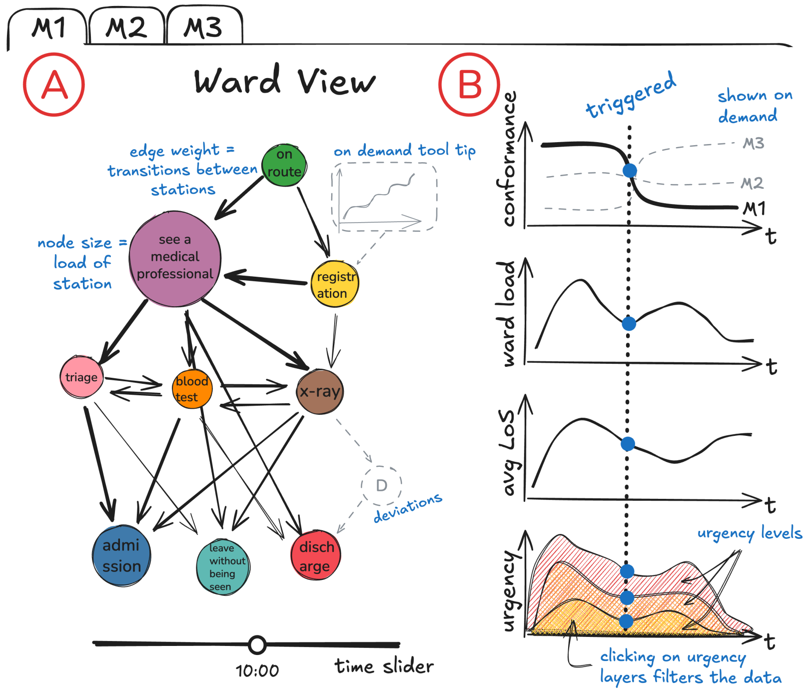

Fig. 3. **VESPA-VIS *Ward View*** comprising (A) the *Patient Flow* view displaying the event-streams flowing into the ER ward, and (B) the *Temporal Facet* view displaying the temporal distribution of relevant facets.

low, medium, high) is displayed as a stacked area graph, allowing an expert to reason about the characteristics of the patients currently putting load on the ward. Additional facets could be displayed in a similar manner in the view, if deemed appropriate for the task at hand.

The *Patient Flow* and *Temporal Facet* views are updated according to three timing strategies: **(1)** at regular pre-defined intervals (e.g., every 10 minutes or 100 events) by default (scheduled), **(2)** if a guard/trigger is activated (triggered), **(3)** upon request of the user (on-demand). The *Ward view* is displayed for all of the available reference models in different tabs. A user can switch between exploring the event-streams against these at any time.

The *Patient View* (cf. Fig. 4) is designed to drill-down into the individual patient event-streams when a need arises. In the *Patient View*, the individual patient streams are displayed as event sequences. Time is displayed on the horizontal axis and the user can toggle relative and absolute time. Patient sequences are sorted along the vertical axis by an *urgency score*. If the computed *urgency score* of a patient exceeds a predefined threshold, a guard triggers, calling attention to the need to prioritize individual patients. The *urgency score* is computed as a distance from a benchmark sequence. We consider three alternative benchmark

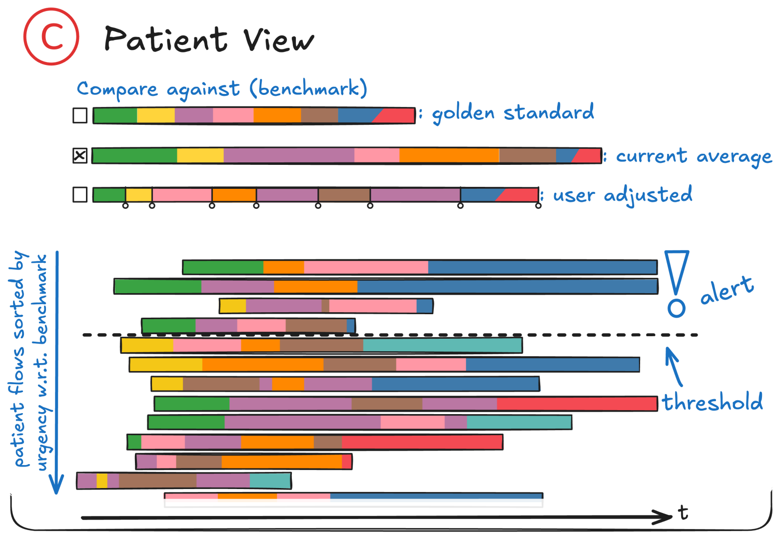

Fig. 4. VESPA-VIS *Patient View* displaying individual patient streams as sequences of events sorted along the vertical axis by an *urgency score*.

sequences in VESPA-VIS: **(1)** The "Golden standard" pre-defined by an expert as an ideal path through the process both in terms of sequence of events and timing. Different ideal sequences can be defined for different times of day or days of the week. **(2)** The "Current average" as a typical patient sequence reflecting the current ordering and average duration of events. **(3)** A "User adjusted" patient sequence where an expert (e.g., ER manager) makes online decisions regarding the target duration of events. The ability to choose between benchmarks to compare against allows the expert user of VESPA-VIS (e.g., ER manager) to flexibly adjust the notion of urgency and control prioritization of patients according to the current situation, their domain knowledge and previous experience.

Together, the *Ward* and *Patient views* allow users to transition smoothly between user roles: monitoring the current situation, reacting on evolving changes, reasoning about possible explanations, and potentially anticipating outcomes.

4 Conclusion and Future Work

The presented VESPA-VIS mockups provide an initial illustration of our research questions. That is, when and how PVA can best support real-time decision-making in a SPM context. The illustration of the approach through the use case indicates fertile ground for further developing the approach, however, there remain a number of further considerations for the approach to be fully realized.

A robust evaluation with carefully planned user studies with representative participant groups will help to assess the effectiveness of PVA for (real-time) decision support. Given the continuous nature of the event-stream, it is natural to expect a need to 'forget' previous event-streams when they are no longer relevant for the current decision making. For example, the straightforward way of simply 'forgetting' patients who have exited the ER through discharge or transferal to another ward, prevents the head of the ER department from comparing the currently observed situation with previously observed situations such as processes occurring on a current New Years holiday day to the processes on the same day in previous years, to identify best practices or simply "what has worked in the past". Identification of 'forgetfulness' thresholds is in itself a complex and multi-faceted problem that requires further work, although prior literature gives hints (e.g., [12, 13]). Although the focus of the approach is to support real-time decision making, the insights gained from the proposed approach present an opportunity to inform process enhancement.

Acknowledgements. This paper resulted from discussions at Dagstuhl Seminar 25152 "Multi-Faceted Visual Process Mining and Analytics".

References

1. van der Aalst, W.M.P.: Process Mining - Data Science in Action. Springer, 2nd edn. (2016)
2. Aigner, W., Miksch, S., Schumann, H., Tominski, C.: Visualization of Time-Oriented Data. Springer, 2nd edn. (2023)
3. Beck, F., Burch, M., Diehl, S., Weiskopf, D.: A Taxonomy and Survey of Dynamic Graph Visualization. CGF **36**(1), 133–159 (2017)
4. Burattin, A.: Streaming process mining. In: van der Aalst, W.M.P., Carmona, J. (eds.) Process Mining Handbook, LNBIP, vol. 448, pp. 349–372. Springer (2022)
5. Burattin, A.: Beamline: A comprehensive toolkit for research and development of streaming process mining. Softw. Impacts **17**, 100551 (2023)
6. Burattin, A., Sperduti, A., van der Aalst, W.M.P.: Control-flow discovery from event streams. In: Proc. of the IEEE CEC. pp. 2420–2427. IEEE (2014)
7. Fekete, J.D., Fisher, D., Sedlmair, M. (eds.): Progressive Data Analysis – Roadmap and Research Agenda. Eurographics Press (2024)
8. Hadlak, S., Schumann, H., Schulz, H.J.: A Survey of Multi-faceted Graph Visualization. In: Eurographics Conference on Visualization. The EA (2015)
9. Jo, J., L'Yi, S., Lee, B., Seo, J.: ProReveal: Progressive visual analytics with safeguards. IEEE TVCG **27**(7), 3109–3122 (2021)
10. Mackinlay, J.: Automating the design of graphical presentations of relational information. ACM Transactions on Graphics **5**(2), 110–141 (1986)
11. Micallef, L., Schulz, H.J., Angelini, M., Aupetit, M., Chang, R., Kohlhammer, J., Perer, A., Santucci, G.: The human user in progressive visual analytics. In: Short Paper Proceedings of EuroVis'19. pp. 19–23. Eurographics Association (2019)
12. Pavan, A., Chakraborty, S., Vinodchandran, N.V., Meel, K.S.: On the feasibility of forgetting in data streams. Proc. of the ACM on Management of Data **2**(2) (2024)
13. Schulz, H.J., Weaver, C.: Transient visual analytics. In: Proc. of EuroVA. Eurographics (2024)
14. van den Elzen, S., Jans, M., Martin, N., Pieters, F., Tominski, C., Villa-Uriol, M.C., van Zelst, S.J.: Towards multi-faceted visual process analytics. Information Systems **133**, 102560 (2025)
15. Van Wijk, J.J.: Views on visualization. IEEE TVCG **12**(4), 421–432 (2006)