TOWARD A SONIFICATION CONCEPT FOR BUSINESS PROCESS MONITORING

Tobias Hildebrandt and Stefanie Rinderle-Ma

University of Vienna, Austria, Faculty of Computer Science
{tobias.hildebrandt, stefanie.rinderle-ma}@univie.ac.at

ABSTRACT

Real-time monitoring of business processes poses several challenges, many of which (like observing process executions and at the same time focusing on other tasks) cannot be tackled with state-of-the-art, visualization-based process monitoring systems. Due to the inherent characteristics of process execution data - it is by definition time-based and sequential - as well as to the typical mode of monitoring business processes (in parallel to other tasks), sonification seems to be the perfect fit to enhance current visualization-based process monitoring with sonification. Therefore, this paper firstly tries to build a foundation by analyzing the task of business process monitoring as well as the data structure of process execution data. Based on these results, existing research that analyzes the usage of sonification techniques for data of similar structure is being studied in order to come to a list of recommendations that can serve as a guideline when building sonifications based on similar data. These recommendations are then being applied to the domain at hand and serve as an input for a first concept of a sonification-based business process monitoring system.

1. INTRODUCTION

It is crucial for most of today's businesses to be able to monitor the executions of their business processes in real time in order to adapt to sudden changes (such as a standstill in production) or to react quickly to undesirable situations (such as a stock running low or the fact that a delivery of goods that are elementary for production is delayed) [1]. Current state-of-the-art business process monitoring typically bases on so-called dashboard- or cockpit-views that aggregate individual process execution events and present those figures in real time, using visualization metaphors such as speedometers [2]. Users who have an interest in monitoring process executions (such as technicians or managers) usually pay attention to these dashboard overviews periodically, while at the same time interrupting other activities they are working on. This of course has the drawback, that these users only learn about situations that are possibly critical to their business when they next decide to look at their monitoring application. Extending existing visual tools for process monitoring with sonification can tackle this problem, as sound can draw the users' attention in case of alerts or other notifications. But we believe sonification can do more for business process monitoring than to convey alerts: it can keep users constantly aware of their processes and even their business' performance by conveying certain operating figures (like the percentage of overdue orders) aurally. Furthermore, we believe that a sonification of all the individual events that occur during process execution (such as the starting and stopping of the processes' individual steps) can enable users to even anticipate possibly business-critical situations before they occur. Due to humans' ability to detect smallest changes in rhythm and sequence it should be possible for users to immediately notice if e.g. activities of a process are being executed in a different order than usually, or to hear that "something is off", e.g. when the time period between two production steps is longer than normal. Furthermore, after listening to their process executions for a while, users might even be able to detect bottlenecks or, more general, room for improvements of the process models itself.

The design of a sonification that aurally conveys all occurring process events will be challenging (there is e.g. a high risk of an information overload), but such a system would probably still be an improvement over the current state of the art in business process monitoring. Such systems, as they are offered by many companies like for example ARIS [3] or IBM [4], have their main focus on presenting aggregated process execution data in a graphical way, while the individual events that occur during execution are only conveyed in textual form. Sonification might be able to alleviate the situation. Process execution data is per definition sequential and time-based, as is the way we perceive sound. Therefore we believe that sonification is the perfect fit for process monitoring. However, to our best knowledge, so far no sonifications have been developed that are explicitly intended for business process monitoring and that take business process-related concepts (such as process models and -instances) into consideration.

This paper therefore approaches the goal of developing such a sonification by firstly analyzing what is important to consider in business process monitoring (Section 2), and what makes business process execution data so special (Section 3). Based on this analysis, publications that deal with sonifications of data that is structurally similar to that of process execution data will be selected from an existing survey [5] in Section 4. This list of related research is then analyzed regarding which of the mappings and sonification techniques the respective authors applied were deemed beneficial (based on formal or informal evaluations of the respective authors) in Section 5, leading to a list of recommendations to take into consideration when developing sonifications for similar data structures. This publication then presents an initial concept of how such a sonification for business process monitoring could be designed (Section 6) and concludes in Section 7.

2. BUSINESS PROCESS MONITORING

A business process is a structured set of tasks or activities that are designed towards reaching a business goal [6]. Business processes have a beginning and an end with clearly defined input and output parameters, while the tasks or activities that the processes consist of can have resources such as users or roles assigned to them. Such a description of the activities that are involved in a business process together with the definition of control and data

flows is called a process model, while each execution of such a process model is called a process instance [6]. Business processes can take a variety of forms, ranging from long running processes whose individual activities are mainly performed manually (such as an insurance claim process) to highly-automated, fast-running processes that contain no human involvement (such as a production process in a highly automated factory). Figure 1 shows a simplified part of a business process of a manufacturing company.

Business Process Management (BPM) is a management approach that covers the complete life cycle of processes, from their initial design (the design phase) over their execution (the operation phase), over the retrospective analysis of historic process execution data (the analysis phase) again to the improvement of the process models based on the findings from the analysis model (the change phase). A particular challenging phase of the business process life cycle is the operation phase, where one crucial task is to monitor currently executed process instances. Companies can have hundreds or more business process models, while there usually are many instances of those process models running at any given moment. As those instance executions can be forced to a stop due to e.g. technical problems, and a high-level overview over all instances of a process can give an accurate overview over a processes' or even a companies' performance, enterprises want to keep track of the executions of their processes, often in real-time. Current state-of-the-art process monitoring systems aggregate individual process execution events and present them with regularly updated visualizations. Occurring alerts and notifications are being presented in textual form. Additionally, many systems provide graphical overviews for all currently running process instance that mark the current status and position of the respective instance. However, even though these systems are typically geared towards presenting high level information on processes, different user groups have different information needs: Technicians or people working on concrete process activities are typically more interested in low-level information, such as individual events that occur or specific error or warning messages. Managers on the other hand are often less interested in the individual events, but want high-level overviews over process performances. Users from both user groups usually pay attention to these dashboard overviews periodically, while at the same time interrupting other activities they are currently working on.

3. BUSINESS PROCESS EXECUTION DATA

Business process execution data is in many ways different from data of many other application domains. Firstly, a significant share of sonifications that have been developed so far, be it for monitoring or for data analysis purposes, base on quantitative data that change over time. Business process execution data on the other hand does in principal base on sequential, semantical and hierarchical instead of quantitative data. Thus, while in many domains "how much" or "how many" are the most important aspects, in business process monitoring it is rather "what" and "when" and "in what order" (often combined with additional information such as "who" or "where"). But business process execution data also differs from data that is at first sight structurally very similar, like program execution data. This chapter will explain the data structure of process executions in more detail and analyze its commonalities and differences to data of similar structure.

At its core, business process execution data consists of event-based, qualitative data. Its foundation are the individual log entries of an execution engine (a software system that is responsible for coordinating the different activities of the processes and monitor the process instances' execution status). Typically, for each event that occurs during process execution, one log entry is created (see bottom of Figure 2).

Such a log entry typically consists of an activity that has been performed or is about to, the event type (usually either starting or stopping of activities), a time stamp and an assigned user. Depending on the execution engine, instead of an activity that has been started or stopped, log entries can also be created for different events, e.g. the modification of variables or the occurrence of errors during execution. Beside this minimum set of information, depending on the company and its system, an arbitrary amount of additional semantic and also quantitative information can be included in the log entries as well. In general, events could be classified into three categories:

- Control flow (e.g. starting/stopping of activities)
- Data flow (changes in variables)
- Warnings/errors

Standards like XES (eXtensible Event Stream) [7] define the data structure of log files. Implicit relational information is included in the individual event logs, as most events occur in the context of a process instance, which in turn is instantiated from a certain process model. To summarize, an event usually entails the following information:

- Event base type (control flow, data flow, warnings/errors)
- Event type (e.g. activity started/stopped, variable changed)
- Relational information (Relation to process, instance)
- Semantic information (e.g. name of the respective activity or variable, name of responsible user and/or department, detailed error or warning message...)
- Related quantitative data

The execution of a process instance usually entails several or (for large process models or processes with many cyclic parts) sometimes even hundreds or thousands of such individual events. Still, most data on the process instance level is of qualitative nature. The data of all instances of a process are often aggregated to instance-spanning, quantitative parameters. Of course, the amount of data that accumulates over a certain period of time varies heavily, depending on the types of business processes a company runs. In labor-intensive, long running processes it may happen that there is only one running instance at a time (if e.g. a small workshop produces only one unit at a time), with only a few events occurring per day. On the other hand, with highly automated manufacturing processes in which e.g. events are triggered based on sensor data, thousands of instances can exist that may possibly lead to thousands of occurring events per second. In such cases, users are probably not interested in monitoring all individual events that occur. Therefore, most state-of-the-art business process monitoring systems allow the definition of instance-and even process-spanning performance measurements, so-called KPIs (Key Performance Indicators). These quantitative KPIs are aggregated over the individual events that occur. They are usually being updated in regular intervals and have the objective to give users a general overview over how busy or how "healthy" their processes are, thus giving an impression on how well business is going. Such quantitative parameters could be calculated for single process instances (like

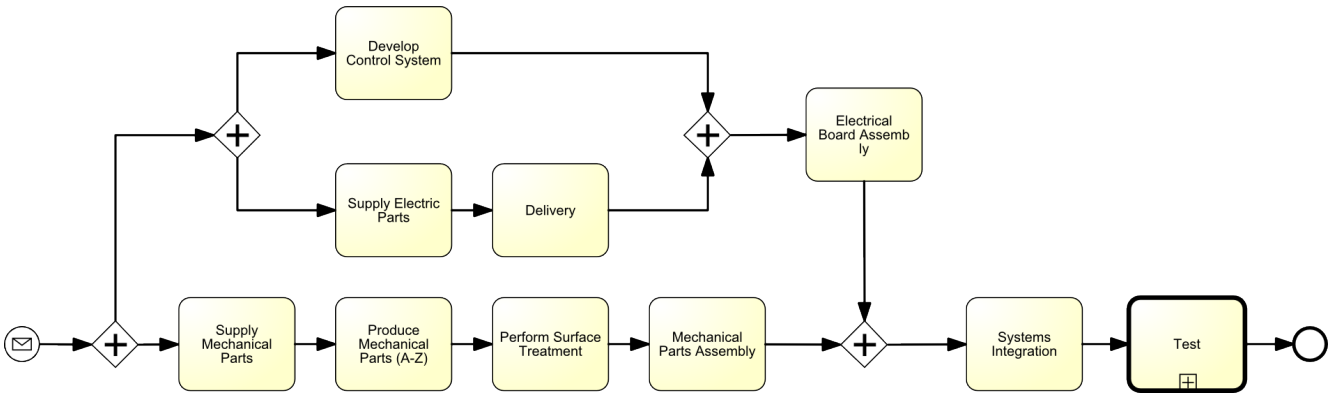

Figure 1: Simplified example of a business process model.

the current execution time versus the average execution time for instances of the respective process), while they are typically calculated either over all instances of a specific process (e.g. the share of instances in faulty states) or over all instances of all processes that a company runs (like e.g. average process instance overdue time).

4. SONIFICATION OF QUALITATIVE, EVENT-BASED DATA

It appears, that so far there has been no research that dealt explicitly with the development of Auditory Displays of business process execution data, a fact that has already been pointed out [8, 9]. There is however research regarding sonification in industrial production processes, which can be regarded as a specific form of business processes. Existing sonfication research in this area however, as e.g. conducted by Gaver et al. [10] does not take a business process-centric perspective, but instead sonifies industrial productions from the viewpoint of the individual machines and work steps. Thus, business process-inherent concepts such as the relation between process models and process instances have not been considered for sonification. However, approaches like the mentioned work by Gaver et al. [10] share with the domain at hand certain attributes like the basing on real-time, event-based qualitative data, which leads to the assumption that many of its results can be transferred to the domain at hand. The same holds true for program execution sonifications, as a process model is very similar to a program code, while one process instance can be compared to one program execution. Both consist of a control flow, a data flow and errors and warnings.

Besides all these similarities, the two domains also have its differences: process execution data seems to be substantially more complex than program execution data: when debugging a program, usually only one instance of that particular program is being executed at a time. For one process model however, up to thousands of parallel running instances (although this may not be the standard case) can exist. Furthermore, companies typically run several processes, which all have their own instances. Parallel programs do possibly posses a certain amount of the hierarchical and structural complexity of process execution data. There have been a few approaches towards sonification in this domain. Parallel program executions could be, depending of the type of parallelism, compared to different instances of the same process which are being run simultaneously. However, users who are monitoring process executions are also interested in instance-spanning, quantitative data. Besides the differences in data structure, the tasks of monitoring programs and processes differ inherently: program code debugging usually consists of short sessions in which the users dedicate their full attention to the task of bug-finding. Process monitoring on the other hand is a task usually performed as a background activity during the whole working day. Table 1 summarizes the different domains.

Table 1: Domains with qualitative, event-based data.

Domain	Simultaneous executions	Type of monitoring
Program executions	No	Short, direct
Parallel program executions	Yes	Short, direct
Business process executions	Yes	Long, peripheral

Thus, even though there seem to be no sonifications so far that base on business process execution data, there seems to be at least sonifications of similar data. It can be expected that many of the results from studies with similar data can be transferred to the domain at hand. A survey of existing approaches to the usage of sonification for the task of process monitoring has already been conducted by Vickers [5]. In the next chapter, those of the sonifications that have exemplarily been presented in mentioned publication will be analyzed in more detail, which base on similar data to that of the domain at hand.

5. LITERATURE ANALYSIS

In this chapter, all sonification projects that have been presented examplarily by Vickers [5] are analyzed in terms of how similar the underlying data structure is compared to the domain at hand. Afterwards, the resulting list of related publications is analyzed in terms of which mappings and techniques were applied, and if they were successful or not, based on formal or informal evaluations of the papers authors (if conducted). Based on these results, a list of recommendations and guidelines for sonifications of similar

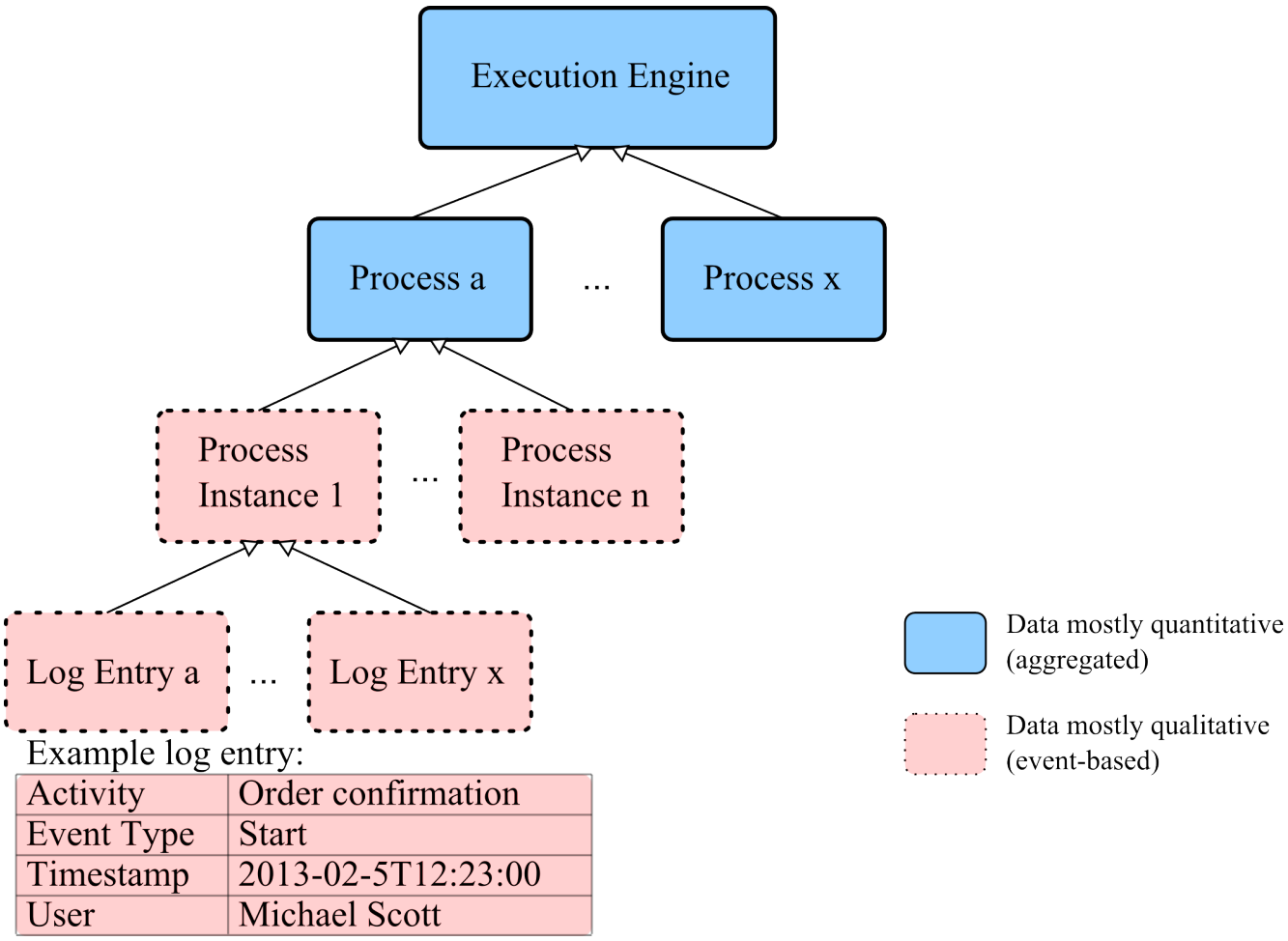

Figure 2: Structure of business process execution data.

data structure will be compiled, whose results will subsequently be transferred to the domain of business process monitoring. The criteria for inclusion in this analysis were firstly, that the data that the respective sonifications base on had to be of primarily qualitative nature and convey occurring events in real time. Additionally, those of the publications presented by Vickers [5] that deal with sonification of parallel programs are also being analyzed in more detail, as they base on data structure that is quite similar to the one at hand (as pointed out in Section 4).

5.1. Results and Findings

The majority of the literature that has been discussed in detail by Vickers [5] bases on sonifications of similar data structure as the one at hand and was therefore included in this literature analysis. The publications base on a variety of application domains, such as industrial monitoring (ARKola [10], Sharemon [11], [12]), home and shared work environments (ListenIn [13], Music Monitor [14], WISP [15], RAVE [16], Workspace Zero [17]), systems for external auditory representations of programs (Caitlin [18, 19, 20], Infosound [21], Logomedia [22], Sonnet [23], ADSL [24], Listen/LSL & Jlisten [25], Program Slices [26]), web server and internet sonification (Peep [27], WebMelody [28]) and interface tasks (SonicFinder [29]). The majority of the publications that were analyzed describe sonifications that convey occurrences of discrete events. One example for such a sonification is the ARKola Simulation by Gaver et al. [10] where occurring events such as spills of liquid are being sonified using Auditory Icons that have a predefined sample length. Process execution data on the other hand also bases on discrete events but also contains activities that have certain durations which are marked by the respective ”activity started” and ”activity finished” events. There are a few sonifications like the CAITLIN project by Vickers et al. [18], where constructs that have a certain duration (like a for-loop) also convey these durations aurally (by using continuous sounds for the whole duration of an ongoing activity). Gaver in his project Sonicfinder [29] even applied sonic metaphors that can give a clue about the remaining duration of an ongoing activity (he applied sounds of a jug being filled with water to sonify an operation systems’ copying procedure). In terms of the selected mapping techniques, two almost equally big fractions can be identified: sonifications that apply mappings of events to Auditory Icons, and sonifications that apply Earcons, which in most cases based on harmonic principles of western music, or musical motifs (short melodic patterns) like e.g. [18]. A few of the analyzed sonifications let the user define

the sounds that are being used, as well as the events that trigger their playback. Some approaches (like [18]) not only aurally convey event occurrences but furthermore tried to convey structural or hierarchical information as well, mostly by using hierarchically structured classes of motifs or timbres.

In the following, based on conducted formal and informal experimentation of the authors of the respective publications, sonification techniques and mappings that have successfully been applied to convey certain concepts will be presented and summarized. In Cohen's OutToLunch [12], a sonification that was based on motifs was deemed more pleasant than an earlier version with Auditory Icons, although this fact has not been formally evaluated. On the other hand Berman [30] concluded (based on user evaluations) that the associations of concrete sounds are easier to recall than those of abstract sounds, and that they therefore should be preferred over musical sounds if they are available, as they entail less cognitive overhead. Vickers [19], applying musical Earcons, further suggested (based on user testing) not only to use melodic constructs, but to include percussion and rhythm in sonifications as well. Furthermore, he states that tonal music should be preferred over direct mappings to frequencies [20].

Different approaches have been taken to differentiate sonically between different constructs. Vickers stated that by using different motifs (short melodic patterns) users are often able to distinguish different constructs but that it is important to at the same time also use different timbres/instruments for the different constructs [18, 20]. He also suggests that motifs should also differ in rhythm and tempo and to convey durations of activities by using sounds that are being played back for the whole duration of an activity (e.g. by using drone sounds) [18]. This concept has also been applied by Francioni et al. [31]. Vickers [18] furthermore specifically suggests to investigate the usage of musical contour (the direction and shape musical notes move in, e.g. a short melody that is rising in pitch) as they might yield better results than motifs that do not take such considerations into account [20]. Francioni et al. [31] on the other hand distinguished different concepts (in this case different processors) by assigning each concept a different timbre. Thus, while the different events that occurred during execution had each a different note assigned to it, those notes were played in the respective instrument for the processor the events occurred in.

Another concept that has been conveyed aurally (e.g. by Vickers [18]) is hierarchical information. Vickers mapped hierarchical information to hierarchical groups of leitmotifs and derived motifs, stating that most users were able to distinguish between the different top-level constructs that have been sonified using leitmotifs, while they had more problems in distinguishing the derived motifs that have been assigned to the respective sub constructs. Further he concluded that in general, the developed sonification for debugging purposes proved especially beneficial for very complex programs, while it was less helpful for debugging simpler constructs.

Vickers also suggested to let users not only decide which instruments the different constructs are being mapped to, but to let users influence the melody creation as well. [20, 32]

5.2. Summary

To summarize the results from the literature analysis, based on the suggestions of the respective authors the following things should be taken into account when designing sonifications for the real-time monitoring of processes that are based on mainly qualitative, event-based data:

- Users should be able to customize the mapping from data to sound.
- If concrete auditory representations for the occurring events are available, the usage of Auditory Icons can yield positive results.
- When occurring events are mapped to Earcons, complex timbres (possibly based on real-world instruments) should be preferred over simple timbres (like sine waves).
- The Earcons should take concepts from the areas of motif design and melodic contours into considerations and adhere to "musical" concepts (such as the western tonal system).
- If motifs are being applied, they should differ not only in pitch, but also in rhythm and intensity.
- Different concepts can be conveyed by using different motifs (possibly hierarchically structured) and/or different timbres.
- In general, rhythm and percussion should be included in sonifications.
- Continuous sounds (such as drones) should be used to convey the duration of ongoing activities.

6. APPLYING THE FINDINGS TO BUSINESS PROCESS MONITORING

Based on the findings of the literature analysis in the last chapter, a first suggestion of how to design a sonification for business process monitoring can be made. In general, the literature analysis suggested that a process monitoring sonification should offer possibilities for the user to customize the mappings from data to sound. This could for example be realized by providing a GUI with menus. It seems furthermore beneficial to offer the user possibilities to select in real time what data is sonified in what level of detail (eg, by offering drop-down menus with various filters and selections). This seems important since, as already mentioned, different user groups (such as technicians or managers) have different information needs. Even though it seems recommendable that the specific mappings should be user-adjustable, in the following, guidelines on how to design an initial prototype for auditory business process monitoring will be presented. These guidelines take into account event occurrences as well as continuous KPI conveyance.

6.1. Individual Events

As one outcome of the literature analysis was that different constructs can be conveyed by melodic Earcons (such as motifs, e.g. following the guidelines of Vickers et al. [19]) and different timbres, it can be expected that those concepts can also be applied to the domain at hand. These motifs could perhaps take into account the concept of melodic contours, as e.g. suggested by Cullen et al. [33]. In principal, process execution data has two orthogonal types of constructs that can be conveyed:

- The hierarchical relations between processes, instances and activities
- The hierarchy of event base types (control flow, data flow, errors/warnings) and the concrete event types (e.g. activity started)

In most cases, users are probably less interested in the event type hierarchy, and more in distinguishing the different individual events. In fact, probably a significant proportion of users

only wants to distinguish between the different warnings and errors that occur during execution, while others may also want to get informed about the starting and stopping of activities or other events. Thus, it is possible to reduce the complexity of the conveyed information by only distinguishing between the different event types without conveying their type hierarchy. This would leave only the hierarchy of processes, instances and activities to be sonified, which could be done by either using hierarchically structured motifs or hierarchically grouped timbres.

As companies typically have several processes that have a number of instances and activities each, users could easily become overwhelmed when trying to convey all those concepts over hierarchically structured timbres. Thus, it seems sensible to let users choose the construct that is most important for them to distinguish, and then convey only this concept (either processes, instances or activities) using timbres that are as different as possible. Thus, when using motifs and timbres to convey the different constructs, two options remain:

1. Convey the different processes, instances or activities over different timbres and the different event-types over distinct motifs
2. Convey hierarchical relations between processes, instances and activities over hierarchical structured motifs, and the different event-types over different timbres

Which of the two options yields better results will have to be evaluated in experiments and probably also depends on the type of processes a company runs and the information needs of the user company, and therefore should be choosable by the user. It seems however, that the motif creation is easier for (1), as it is probably easier for users to associate a specific Earcon or Contour Motif with an event type (like ”activity started”) than with an abstract concept (like e.g. ”process instance 4”). As an example, a Contour Motif consisting of a few notes that rise in pitch could signify that an activity has started, while a falling pitch could mean that an activity has finished.

There may also be cases where users are only interested in distinguishing one of the two concepts, and may thus choose to apply different timbres as well as motifs to distinguish this concept. For example, for some users it may be especially important to e.g. distinguish between a variety of events (e.g. between different warning/error types) while it may be less important to them, in which process or instance they occur. In this case, the user may decide to map the event type hierarchy to hierarchical motifs that also are played in different timbres.

If the users e.g. opt for (2), all the individual events that occur could then be conveyed by playing the specific motif of the instance or the activity. For both, (1) and (2), a possibility to convey the fact that an activity is ongoing (i.e that the ”activity started” event has occurred), could be to loop the motif that is assigned to this event until the respective ”activity finished” has occurred. Of course, this method could potentially be annoying in cases of long-running activities and is only beneficial for a limited number of parallel activities.

6.2. Quantitative KPIs

As already discussed, aggregated quantitative parameters are an essential means to monitor the performance of business processes. Techniques from Parameter Mapping are an established means to convey such continuous, quantitative data, which is why it can be assumed that such techniques will be suitable for the conveyance of KPIs as well.

For KPIs on the process level, one solution could for example be to create a different continuous drone sound for each process model and map different process level KPIs to acoustic properties of these drones. If possible, the same concepts that are used to distinguish between the different processes for the conveyance of event occurrences could also be used for those drones. If, for example, for events of different processes different grouped timbres (like e.g. stringed instruments) are used, the different KPI drones for the different processes could also base on those timbres (e.g. by using a drone that also sounds like a stringed instrument). If a company runs too many different processes to enable a discrimination over different drones, a possible solution would be to use only one or a low number of drones and map selected process-spanning KPIs to acoustic properties of these drones. Instance level parameters (such as e.g. variable values that are of particular interest to the user) could for example be mapped to acoustic properties of all motifs that are being played for a particular instance.

6.3. Design Concept

In general, an auditory business-process-monitoring-solution could be used to give users a constant overview over the performance of process executions. The users should have the possibility to adjust the systems to their preferred granularity level. Some users might only want to get informed about errors or certain alerts, while others also want to hear constant sonifications of various KPIs. But there might also be users who want to hear a sonification of all events that occur for certain or even for all processes or instances.

It can be expected that in companies that run processes in which thousands of events per minute occur, a sonification of all individual events would not be very helpful and interesting to the user (at least not in the form suggested in this chapter), while for processes that only have a handful events per day (such in processes whose tasks are mainly executed manually) such a sonification might be beneficial.

Thus, it is recommendable to design the system in such a way that it is flexible enough to offer different modes of conveying event occurrences in order to adjust to different data densities that exist for different processes. For example, de Campo’s Sonification Design Space Map [34] suggests that a sonification that bases on individual notes/motifs (such as the one suggested in this paper) only works for a limited number of events per second (according to the Campo, around 10 events per second and stream). This number will probably in many companies not be exceeded, it might be however for highly automated processes. For such processes, grain clouds based on granular synthesis, as suggested by the Design Space Map, might be a better option than the usage of notes or motifs. A possible solution could be a system that automatically switches between different modes of aurally conveying event occurrences, depending on how many events currently occur per second or minute. Furthermore, even for cases that are suitable to convey individual event occurrences using e.g. motifs, it probably makes sense in terms of perception to only play a very limited number of notes simultaneously (if any), and instead queue occurring events in order to play them sequentially (perhaps starting with urgent events such as alarms).

In general, the Auditory Display should be designed in such a way that during ”normal operation”, i.e. when there are no excep-

tional situations or sudden changes during process execution, users should not be disturbed by the sonification and be able to concentrate on their main task. However, in cases of sudden/unexpected changes or alarm situations, the sonification should be able to direct the users' attention to the GUI of his execution engine, where he can use visual and textual information (e.g., in case an alert occurred, a detailed error description) to analyze the root cause of the situation.

7. CONCLUSION/FUTURE WORK

As we have seen, business process monitoring is an activity that could benefit from sonification. Even though there is a considerable amount of research concerning sonifications of data from various disciplines, the data that aggregate during the execution of business processes is inherently different from that of many domains sonification has so far been applied to. An analysis of related literature has revealed that there exists a variety of sonifications for domains whose data structures have similar properties (event-based, qualitative data) as process execution data. Based on this analysis, a list of recommendations and guidelines for sonifications of similar data structure has been compiled. These recommendations have been transferred to the domain at hand, leading to an initial concept for a sonification for business process monitoring. An important next step would be to conduct user studies in different domains, such as virtual factories within the EU FP7 project ADVENTURE (http://www.fp7-adventure.eu/), in order to find out if the suggested concepts can successfully be applied for building sonifications in this domain. Specific attention should be paid to the way users can customize the sonifications and apply different filters and options to filter the data in real time.

An important aspect will be to design user studies in such a way that they recreate realistic working conditions, for example by drawing the users attention to a main task while letting them listen to a background sonification of process executions. A few of the challenges that have to be tackled in order to develop such a sonification are to find out if event occurrences should be conveyed using motifs (possibly based on principles of melodic contours) or Auditory Icons, or if this decision should be left to the users. Furthermore, it will be interesting to see if users will be able to distinguish between different orthogonal hierarchies/constructs that are mapped to different motifs as well as timbres, and especially between how many of those. Lastly, it has to be found out how the continuous background sonification of quantitative KPIs and the sporadically conveyed event occurrences can be combined in an optimal way.

8. ACKNOWLEDGMENT

This work was partially supported by the Commission of the European Union within the ADVENTURE FP7-ICT project (Grant agreement no. 285220).

9. REFERENCES

[1] P. Kung, C. Hagen, M. Rodel, and S. Seifert, "Business process monitoring measurement in a large bank: challenges and selected approaches," in *Database and Expert Systems Applications, 2005. Proceedings. Sixteenth International Workshop on*, Aug. 2005, pp. 955 – 961.

[2] W. W. Eckerson, *Performance Dashboards: Measuring, Monitoring, and Managing Your Business*. John Wiley & Sons, Oct. 2010.

[3] "ARIS process performance manager." [Online]. Available: http://www.softwareag.com/corporate/products/aris_platform/aris_controlling/aris_process_performance/overview/default.asp

[4] "IBM - business activity monitoring (BAM) software - IBM business monitor - software." [Online]. Available: http://www-01.ibm.com/software/integration/business-monitor/

[5] P. Vickers, "Sonification for process monitoring," in *The Sonification Handbook*, T. Hermann, A. Hunt, and J. G. Neuhoff, Eds. Logos Berlin, Dec. 2011, pp. 455–492.

[6] M. Weske, *Business Process Management: Concepts, Languages, Architectures*. Springer, 2007.

[7] H. M. W. Verbeek, J. C. A. M. Buijs, B. F. v. Dongen, and W. M. P. v. d. Aalst, "XES, XESame, and ProM 6," in *Information Systems Evolution*, ser. Lecture Notes in Business Information Processing, P. Soffer and E. Proper, Eds. Springer Berlin Heidelberg, Jan. 2011, no. 72, pp. 60–75. [Online]. Available: http://link.springer.com/chapter/10.1007/978-3-642-17722-4_5

[8] T. Hildebrandt, S. Kriglstein, and S. Rinderle-Ma, "Beyond visualization: On using sonification methods to make business processes more accessible to users," in *18th International Conference on Auditory Display (ICAD 2012)*. Georgia Institute of Technology, June 2012, pp. 248–249.

[9] ——, "On applying sonification methods to convey business process data," in *CaISE 2012 Forum*, ser. CaISE Forum. CEUR, 2012. [Online]. Available: http://eprints.cs.univie.ac.at/3449/

[10] W. W. Gaver, R. B. Smith, and T. O'Shea, "Effective sounds in complex systems: the ARKOLA simulation," in *Proc. of the SIGCHI conference on Human factors in computing systems: Reaching through technology (CHI'91)*. ACM, 1991, pp. 85–90.

[11] J. Cohen, "Monitoring background activities," in *Auditory Display*, G. Kramer, Ed. New York: Addison Wesley Longman, 1992, pp. 499–531.

[12] ——, "Out to lunch: Further adventures monitoring background activity," in *Proceedings of the 2nd International Conference on Auditory Display (ICAD94)*, G. Kramer and S. Smith, Eds. Santa Fe, NM, U.S.: International Community for Auditory Display, 1994, pp. 15–20. [Online]. Available: Proceedings/1994/Cohen1994a.pdf

[13] C. Schmandt and G. Vallejo, "Listenin to domestic enviroments from remote locations," in *Proceedings of the 9th International Conference on Auditory Display (ICAD2003)*, E. Brazil and B. Shinn-Cunningham, Eds. Boston, USA: Boston University Publications Production Department, 2003, pp. 220–223. [Online]. Available: Proceedings/2003/SchmandtVallejo2003.pdf

[14] Q. Tran and E. Mynatt, "Music monitor: Ambient musical data for the home," in *Extended Proceedings of HOIT: Home Oriented Informatics and Telematics 2000*, 2000, pp. 85–92.

[15] F. Kilander and P. Loennqvist, "A whisper in the woods - an ambient soundscape for peripheral awareness of remote processes," in *Proceedings of the 8th International Conference on Auditory Display (ICAD2002)*, R. Nakatsu and H. Kawahara, Eds. Kyoto, Japan: Advanced Telecommunications Research Institute (ATR), Kyoto, Japan, 2002. [Online]. Available: Proceedings/2002/KilanderLoennqvist2002.pdf

[16] W. Gaver, T. Moran, A. MacLean, L. Lvstrand, P. Dourish, K. Carter, and W. Buxton, "Realizing a video environment: EuroPARC's RAVE system," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '92. New York, NY, USA: ACM, 1992, pp. 27–35.

[17] B. Eggen, K. van Mensvoort, D. Menting, E. Vegt, W. Widdershoven, and R. Zimmermann, "Soundscapes at workspace ZeroDesign explorations into the use of sound in a shared environment," in *Proc. Sixth Int. Conf. on Pervasive Computing*, 2008.

[18] P. Vickers and J. Alty, "CAITLIN: a musical problem auralisation tool to assist novice programmers with debugging," in *Proceedings of the International Conference on Auditory Display ICAD96*, Palo Alto, 1996.

[19] P. Vickers and J. L. Alty, "Towards some organising principles for musical program auralisations," in *Proceedings of the 1998 international conference on Auditory Display*, ser. ICAD'98. Swinton, UK, UK: British Computer Society, 1998, p. 2929. [Online]. Available: http://dl.acm.org/citation.cfm?id=2227605.2227634

[20] ——, "Musical program auralisation: a structured approach to motif design," *Interacting with Computers*, vol. 14, no. 5, pp. 457–485, Oct. 2002.

[21] D. Sonnenwald, B. Gopinath, G. Haberman, I. Keese, W.M., and J. Myers, "InfoSound: an audio aid to program comprehension," in *System Sciences, 1990., Proceedings of the Twenty-Third Annual Hawaii International Conference on*, vol. ii, Jan. 1990, pp. 541 –546 vol.2.

[22] C. J. DiGiano, R. M. Baecker, and R. N. Owen, "LogoMedia: a sound-enhanced programming environment for monitoring program behavior," in *Proceedings of the INTERACT '93 and CHI '93 conference on Human factors in computing systems*, ser. CHI '93. New York, NY, USA: ACM, 1993, pp. 301–302.

[23] D. H. Jameson, "The run-time components of sonnett," in *Proceedings of the 2nd International Conference on Auditory Display (ICAD94)*, G. Kramer and S. Smith, Eds. Santa Fe, NM, U.S.: International Community for Auditory Display, 1994, pp. 241–250. [Online]. Available: Proceedings/1994/Jameson1994.pdf

[24] D. S. Bock, "ADSL: an auditory domain specification language for program auralization," in *Proceedings of the 2nd International Conference on Auditory Display (ICAD94)*, G. Kramer and S. Smith, Eds. Santa Fe, NM, U.S.: International Community for Auditory Display, 1994, pp. 251–256. [Online]. Available: Proceedings/1994/Bock1994.pdf

[25] D. Boardman, G. Greene, V. Khandelwal, and A. Mathur, "LISTEN: a tool to investigate the use of sound for the analysis of program behavior," in *Computer Software and Applications Conference, 1995. COMPSAC 95. Proceedings., Nineteenth Annual International*, Aug. 1995, pp. 184 –189.

[26] L. I. Berman and K. B. Gallagher, "Listening to program slices," in *Proceedings of the 12th International Conference on Auditory Display (ICAD2006)*, C. F. A. D. N. E. Tony Stockman, Louise Valgerur Nickerson and D. Brock, Eds. London, UK: Department of Computer Science, Queen Mary, University of London, UK, 2006, pp. 172–175. [Online]. Available: Proceedings/2006/BermanGallagher2006.pdf

[27] M. Gilfix and A. L. Couch, "Peep (the network auralizer): Monitoring your network with sound," in *Proceedings of the 14th USENIX conference on System administration*, ser. LISA '00. Berkeley, CA, USA: USENIX Association, 2000, pp. 109–118. [Online]. Available: http://dl.acm.org/citation.cfm?id=1045502.1045526

[28] M. Barra, T. Cillo, A. De Santis, U. Petrillo, A. Negro, and V. Scarano, "Multimodal monitoring of web servers," *Multimedia, IEEE*, vol. 9, no. 3, pp. 32 –41, Sept. 2002.

[29] W. W. Gaver, "The SonicFinder: an interface that uses auditory icons," *Hum.-Comput. Interact.*, vol. 4, no. 1, pp. 67–94, Mar. 1989.

[30] I. Berman, Lewis, "Program comprehension through sonification," Ph.D. dissertation, Durham University, 2011. [Online]. Available: http://etheses.dur.ac.uk/1396/

[31] J. M. Francioni, L. Albright, and J. A. Jackson, "Debugging parallel programs using sound," *SIGPLAN Not.*, vol. 26, no. 12, pp. 68–75, Dec. 1991.

[32] P. Vickers and J. L. Alty, "Musical program auralization: Empirical studies," *ACM Trans. Appl. Percept.*, vol. 2, no. 4, pp. 477–489, Oct. 2005.

[33] C. Cullen and E. Coyle, "Musical pattern design using contour icons," vol. International Computer Music Conference Proceedings. Louisiana, USA: Ann Arbor, Michigan: Scholarly Publishing Office, University of Michigan University Library, 2006. [Online]. Available: http://arrow.dit.ie/dmccon/21

[34] A. de Campo, "Toward a data sonification design space map," in *Proceedings of the 13th International Conference on Auditory Display (ICAD2007)*, G. P. Scavone, Ed. Montreal, Canada: Schulich School of Music, McGill University, 2007, pp. 342–347. [Online]. Available: Proceedings/2007/deCampo2007.pdf