*Erschienen in: Visualization and data analysis 2011 : 24 - 25 January 2011, San Francisco, California, United States / Pak Chung Wong ... (eds.). - Bellingham, Wash. : SPIE [u.a.], 2011. - 78680K. - (Proceedings of SPIE ; 7868). - ISBN 978-0-8194-8405-5*

# Visual Pattern Discovery in Timed Event Data

Matthias Schaefer[a], Franz Wanner[a], Florian Mansmann[b], Christian Scheible[a], Verity Stennett[b] , Anders T. Hasselrot[b] and Daniel A. Keim[a]

[a]University of Konstanz, Konstanz, Germany
[b]Lloyds Banking Group, Birmingham Midshires, England

## ABSTRACT

Business processes have tremendously changed the way large companies conduct their business: The integration of information systems into the workflows of their employees ensures a high service level and thus high customer satisfaction. One core aspect of business process engineering are events that steer the workflows and trigger internal processes. Strict requirements on interval-scaled temporal patterns, which are common in time series, are thereby released through the ordinal character of such events. It is this additional degree of freedom that opens unexplored possibilities for visualizing event data.

In this paper, we present a flexible and novel system to find significant events, event clusters and event patterns. Each event is represented as a small rectangle, which is colored according to categorical, ordinal or interval-scaled metadata. Depending on the analysis task, different layout functions are used to highlight either the ordinal character of the data or temporal correlations. The system has built-in features for ordering customers or event groups according to the similarity of their event sequences, temporal gap alignment and stacking of co-occurring events. Two characteristically different case studies dealing with business process events and news articles demonstrate the capabilities of our system to explore event data.

**Keywords:** Event Data, Visual Analytics, Information Visualization

## 1. INTRODUCTION

Temporal events occur in an extremely wide range of applications in business, government, and science. While some of these events can be aggregated over time in a meaningful way and thus be presented in time series visualizations, other application scenarios require each event to be visible. In addition to that, events often do not uniformly spread over time, but tend to be strongly biased. If any or both of these two characteristics are in the data, time series visualizations typically degrade, which means that a lot of display space is wasted or/and not all events can be displayed due to an overlap problem.

To systematically study event data we first define some related basic terminology for events, their properties and associated analysis tasks and then outline our solution to the above problems.

**Event:** *An event is a single, time-stamped item.*

We consider a data point in time as an event, which can be a time-stamped news article or a system event. In this regard we coincide with the definition in EventSummarizer[16] or Mannila et al.[19] Galton and Augusto call such kind of event an atomic event.[8] Guralnik and Srivastava define an (atomic) event as a change of behavior of a dynamic phenomenon.[10] For our visualization only the change of the time-reference of an event is relevant.

Further author information:
Matthias Schaefer: E-mail: schaefer@dbvis.inf.uni-konstanz.de, Telephone: (+49) 07531 884794
Franz Wanner: E-mail: wanner@dbvis.inf.uni-konstanz.de, Telephone: (+49) 07531 883077
Florian Mansmann: E-mail: Florian.Mansmann@uni-konstanz.de, Telephone: (+49) 07531 883070
Christian Scheible: E-mail: Christian.Scheible@uni-konstanz.de, Telephone: (+49) 07531 882410
Verity Stennett: E-mail: VerityStennett@BirminghamMidshires.co.uk, Telephone: (+44) 0190 2325526
Anders T. Hasselrot: E-mail: AndersHasselrot@BirminghamMidshires.co.uk, Telephone: (+44) 0190 2325526
Daniel A. Keim: E-mail: Daniel.Keim@uni-konstanz.de, Telephone: (+49) 07531 883161

### 1.1 Properties of Event Data

Different event data sets display different properties. For a more systematic analysis, they are therefore briefly categorized in this subsection.

**Event Sequence:** *An event sequence is a set of events that are ordered in time.*

**Event Episode:** *An event episode is a set of events that are time-stamped.*

In[19] there is a distinction between event sequences and event episodes. We also want to use these notions but we comprehend them in another way. An event sequence is a set of events that are ordered in time. Thereby, the ordering is the important property. Whereas an event episode is a set of events that are time-stamped and therefore the distance between the atomic events matters.

Under the assumption that every event has an assigned value for some dimensions of its metadata, event data can be further refined into a) *time-synchronous event data*, in which an accurate time-stamp is important, b) *ordinal event data*, where the ordering of the events according to time or metadata plays an important role, c) *aggregateable event data*, which can be summarized for a particular interval, and d) *hierarchical event data*, where the grouping is defined based on a hierarchical structure in the meta data.

### 1.2 Analysis Tasks

To foster a better understanding of analysis tasks for event data, we define the terms significant event, event cluster, and event pattern.

**Significant Event:** *A significant event is a single event that is interesting for some reason.*

**Event Cluster:** *An event cluster is a set of events that are considered as being similar to each other. This may, but not necessarily, include similarity in time.*

**Event Pattern:** *An event pattern is an event sequence or episode that shows some interesting regularity with respect to certain properties.*

Our specific visualization is designed to support an analyst in his task to search for event clusters, event patterns and significant events. Other work, such as[19] focused on finding frequent episodes. An event pattern in our definition is a sequence or an episode that shows some interesting regularity with respect to a certain property. In particular, our first case study deals with business process events in fraud detection where the ordinal character of the events is of importance. In this case, the analysis task is to find event sequences.

On the one hand, our first case study deals with the ordinal character of events in a fraud detection scenario. In this case, the analysis task is to find event patterns in event sequences. Our second case study, on the other hand, is about sentiment analysis in news blogs. Hereby, we consider time-synchronized event episodes and search for clusters and patterns therein. Significant events are a minor aspect of this paper, when dealing with classes of business process events and identifying important news events that explain a certain development. While the first case study builds upon a strict ordering of the events, we slightly release the strict ordering in the news analysis scenario since sorting of news events within one day makes event patterns more salient.

### 1.3 Summary

In this paper, we propose a visual analytics system specifically targeting event data. This novel visual analytics system can be used to analyze events, event sequences and event episodes by representing them both compact and without information loss. Through a visual interface, the system supports the analyst in the task to identify significant events, event clusters, and event patterns. Events are represented through colored rectangles, which are then ordered according to a layout function. In particular, the ordering is determined by the analysis tasks, resulting in ordinal or time-synchronous event visualizations. The system is supplemented by an automated ordering, which places similar event groups next to each other in order to support correlation analysis. Further features are temporal gap alignment and stacking of co-occurring events. This paper is structured as follows, Section 2 presents related work about time and event data analysis, Section 3 presents our systems, Section 4 details two case studies, Section 5 discusses the benefits of our system, and Section 6 concludes our work.

# 2. RELATED WORK

Traditionally, data with temporal aspects was analyzed in so-called time series visualizations, which are discussed in the following subsection. Then related work in the relatively young field of visual event analysis will be outlined.

## 2.1 Time Series Visualization

Time series are an important type of data encountered in almost every application domain. The field has been intensely studied and received considerable research attention, especially with respect to financial and business applications.[2,3,14,15] Concerning particular analysis tasks, not only highlighting patterns is an important aspect, but also arrangement of multiple time series to support comparison between several monitored items as studied in.[11] Hochheiser and Shneiderman's *Time Searcher* system[13] uses traditional line graphs, which can be analyzed using the dynamic query interface. It includes specification of ranges of values and time intervals, query-by-example, queries over multiple time-varying attributes, query manipulation, pattern inversion, similarity search, and graphical bookmarks. Other application scenarios deal with the problem of identifying patterns on larger time scales by using traditional metaphors for visualization, such as clocks[5,27] or calendars.[4,25] Yet another common approach to cope with time are small multiples (e.g.,[20,22]) or multi-resolution representations.[12,18,21] A broader overview of visualization methods for time-oriented data can be found in.[1] A lot of this work in time series visualization only represents aggregated values, whereas each atomic event is important in many applications of timed event data. In sentiment analysis of news, for example, an averaged sentiment score has only little meaning since it can hide important characteristics of the underlying event data, such as a controversial debate with very negative and positive opinions at the same time.

## 2.2 Visual Event Analysis

Event-based systems have a broad application range in research and the industry with an application scope varying from genome research to business intelligence and analysis. *Event Tunnel*[24] is one such event analysis systems for business processes. In these tunnel plots, the inner circles contain old events, whereas new events are plotted larger on the outer circles. A single business process is thus represented through a chain of connected dots from the inner to the outer circles. The angular axis can be used for assigning an additional data dimension of the business process. Alternative layouts are tunnel plots with two assignable axis and scatterplots. Other variables of an event, such as the type, status, etc. can be encoded using the dots' color and border, or by altering the shapes of the event representations. *WireVis*[7] introduces a system which also deals with fraud detection in the bank sector like one of our case study does. The authors present a tool with different visualizations based on identifying specific keywords within wire transactions. It is very useful for advanced investigators in the bank who are able to detect accounts and transactions with suspicious behavior. The tool was implemented to deal with this very specific task and it was planned to integrate it in the bank's daily work flow. *Gapminder*[23] comes quite close to what we understand by an event analysis system. Its animated scatterplot visualization displays a snapshot of two preset variables for each country in each time interval. Single countries can be marked in order to track the event episode of a country's development over time. This is visually encoded through a number of connected dots in the scatterplot. While old events of unmarked countries disappear in the animation, the marked country's events are maintained throughout the animation. The geographic research community defines events through both temporal and spatial references, which results in special requirements for geographic visualization. One example in this field is the space-time cube,[9] which maps spatiotemporal events using geographic coordinates on the first two dimensions and time on the third dimension. Atomic events are then connected with connecting lines and form event episodes. Animation can be used as an alternative visual representation as shown for telecommunication network and service events in *SWIFT-3D*.[17] We discard animation as a visualization option for event data since it is hard to track large quantities of events appearing and disappearing. While these systems and publications have demonstrated some of the potential that visual event analysis can have in specialized application domains, specific visualizations for that vast amount of data are still "in their infancy".[24] Our paper therefore presents a more general approach for timed event data, which we demonstrate on two characteristically different event data sets.

## 3. VISUAL ANALYSIS OF EVENT DATA

With the availability of large storage devices, huge memory chips and multi-core CPUs, computers for capturing and storing massive amounts of data have become an affordable commodity even for small businesses. Likewise, running resource intensive data mining algorithms is mostly not a problem anymore. However, drawing the correct conclusions and gaining insight into raw data and results of data mining algorithms is still an essential and often unsolved challenge. Visual analytics aims at bridging this gap between automated analysis techniques and the human analyst by combining the former with human-interpretable visual interfaces.

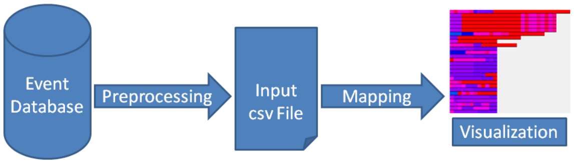

Figure 1. System pipeline with a preprocessing and a mapping step to achieve a high flexibility of the approach.

In this section we demonstrate how our system supports the interaction between the data mining and the visualization techniques on the way from data in the database to new insights. By solving real application problems using both automated and visual techniques, we will demonstrate how significant events, clusters and patterns can be identified.
Figure 1 shows our system pipeline, in which events and their associated metadata are stored in a database. There are two user-driven processes: a) first the preprocessing step defines which attributes of the dataset are used for grouping event into sequences or episodes, and b) the visual mapping step, which assigns visual properties of the representation to dimensions of the event data. With this approach we receive a high flexibility in processing and visualizing different kinds of event data. Note that at this stage the system is developed for advanced users who have domain knowledge and the use cases show its successful application for visual pattern discovery in event data.

### 3.1 Data Preprocessing and Mapping

The basis for the preprocessing algorithm is data which is stored in a database. As shown later this provides great flexibility for creating data for the preprocessing step with database tools and methods. The strength of this approach is that we can use solid database functions, such as ordering, filtering etc. In the system's preprocessing step the relevant columns are chosen and the data is aggregated and transformed by the system into input files for the visualization. It is also possible to reload the preprocessed data into the database, processing the data in the database and setting new flags, which can be visualized afterwards.
A sequential processing of the data allows us to deal with large volumes of data. The data is aggregated for each entry, which is user defined like bank accounts or news feeds/entities in our examples. The resulting file contains only the entries and their related events in the lines in a flat-file csv format. This is used as input for the mapping step to create the system's visual output and allows a fast processing of large amounts of data in the mapping step, too. Through this sequential processing in both steps, the preprocessing and the mapping, limitations only depend on the assigned memory. We assigned 1.5 GB and processed data with 550000 entries and up to 1000 events each, which lasts about 1 minute for the preprocessing and 30 seconds for the mapping (Intel Core 2 Duo SP9400 (2.4 GHz, 1066 MHz, 6 MB Second Level Cache)).

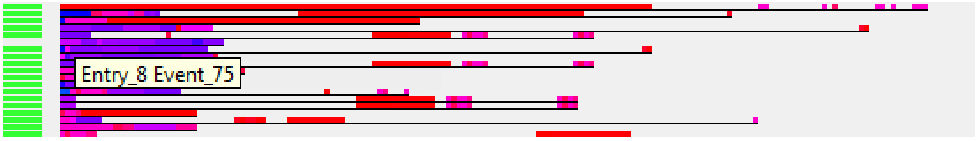

Figure 2. Visual analysis system for event data: Unordered entries.

### 3.2 Visual Analysis System

The system's visual output is shown for the 20 first entries of an event database and their related events in Figure 2. Each line starts with a flag (green or no flag in this example) and represents an entry and its related events. The events are colored according to their defined value. This flexible user-controlled mapping can be easily adjusted to the application and task. For coloring, we have implemented several different color maps, so the most convenient coloring scheme can be chosen for a specific analysis task. Hovering the mouse over an event in the visualization triggers a yellow box with text describing the event as shown in Figure 2. The displayed text can be defined flexible in the preprocessing step using metadata from the database. In addition to that the user can add special flags to the entries for faster identification. In Figure 2 all entries except the sixth one have a flag, which can be seen by the green coloring at the beginning of each line. Flags can be defined easily and flexible in any number in the preprocessing step and help the user to classify the entries.

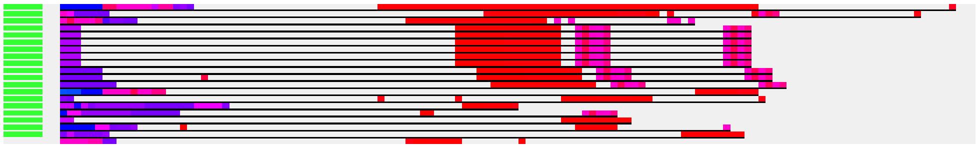

Figure 3. Visual analysis system for event data: Ordering and clustering of entries according to a similarity measure.

### 3.3 Advanced Features

On top of the basic visual analysis system we implemented advanced features which support the user in his visual analysis task. First of all we provide ordering and clustering of the entries based on similarity of the event patterns. Figure 3 has the same data basis as Figure 2 but an ordering step is included in the preprocessing algorithm, which groups together entries with similar event patterns. Again the first 20 results are shown.
Events with the same patterns are clustered together. Another effect is that entry 6 without the green flag as shown in Figure 2 is not in the first 20 entries in Figure 3 anymore. This is because of the dissimilarity of its event pattern to the others. Therefore the last entry with no green flag appears in the result set of Figure 3. This feature helps the user to find entries with similar event data but different flags respectively classes.

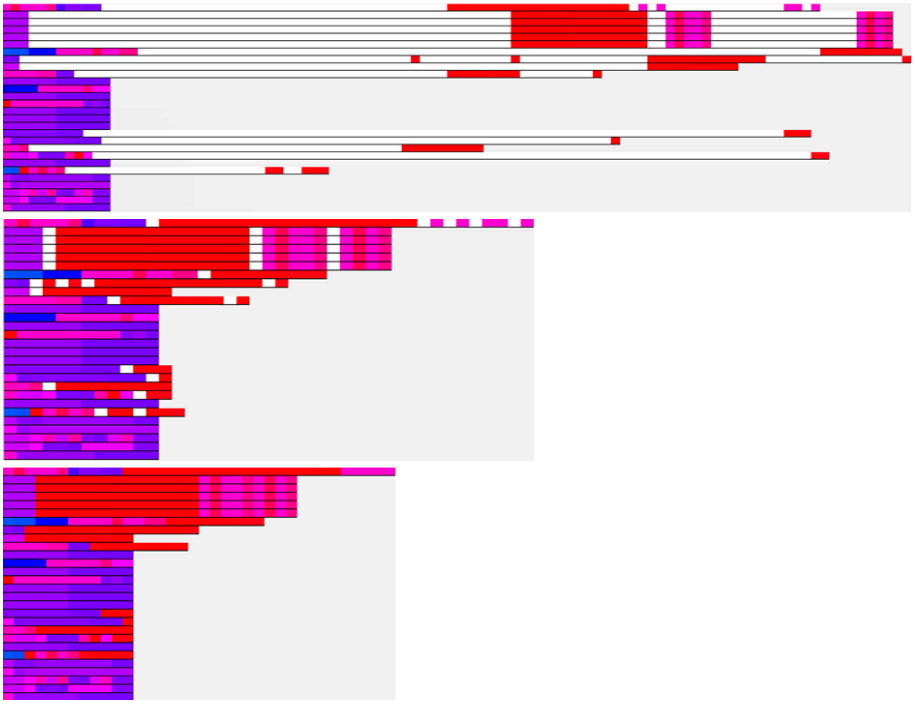

Figure 4. Three different alignments strategies to deal with temporal gaps in event data. Top: visualize all gaps, middle: visualize only one gap, botton: visualize no gaps.

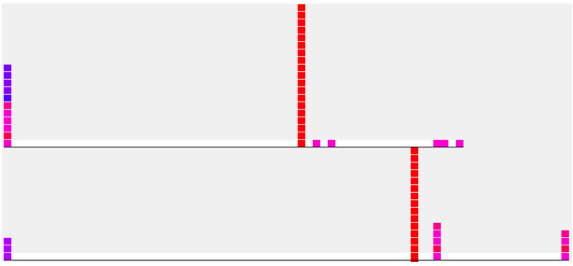

Figure 5. Vertical alignment with stacked events on top of each other when occuring at the same point in time.

Another feature of the system are different alignments of the events. This is important because, as stated before, event data often does not uniformly spread over time, but tend to be strongly biased. For dealing with this problem we provide the user three different options to handle temporal gaps between events. Figure 4 shows them: The top visualization shows one gap for each point in time where no events occurs. The middle visualization reduces this sequence of gaps to only one gap, independent of the sequence's length. The bottom visualization excludes gaps completely.

Stacked events in y-direction are an alternative alignment to deal with the occurrence of more than one event at the same time. So far the previous shown visualizations placed all events of one entry one after each other in one line. This leads to lines with arbitrary length. Figure 5 shows an alternative approach, in which all events at the same point in time are stacked over each other. This is very useful in some applications, since it conveys additional information, such as that many events occured in one day.

# 4. CASE STUDIES

In this section we show the capabilities of our tool on the basis of two characteristically different event data sets. The first case study is about fraud detection in a bank's database, where events are defined as system alerts triggered by customer behavior. In this scenario, we analyze event sequences since more importance is given to the ordering in time than to the exact time-stamps. In the second case study we analyze the sentiment of RSS news postings about the U.S. Presidential Election in 2008. Hereby, the absolute temporal reference of such event episodes are an integral part of the analysis resulting in a different layout of the events in the visualization.

## 4.1 Fraud Detection in a Bank's Database

The first application area is fraud detection in mortgage accounts. To show the system's effectiveness we brought in experts in the operational, strategy and specialist fraud areas who could recommend how best to rank the data and assist in identifying real fraud cases in the event data. All cases were exposed by combining the visualizations with user input and obtaining additional data from the bank's database.

### 4.1.1 Data Set

Data was extracted from a stand alone fraud database used by the bank's fraud teams and combined with internal customer application and performance data. The fraud database contains external information in the form of rules that indicate the possibility of fraud and flags identifying whether applications were investigated internally and found to be fraud or clear. The internal data brought in includes application data such as name, date of birth, bank account details, address information and third party details such as solicitors and brokers. These details are used to rank the event data for visualization. Internal data clarifies if a mortgage has completed successfully, it highlights whether post completion any elements associated with fraud have become apparent and sets out how the mortgage is being maintained, i.e. whether the borrower has fallen behind on their mortgage repayments, if they have defaulted (3+ missed monthly payments) or if the property has been repossessed.

Once the data is collated and ranked appropriately it is read into our system, this data can sometimes include hundreds of rows and several columns per application. The strength of our system is to condense, group and visualize both fixed and time series information on customers in one compact image; allowing the user to identify suspicious individuals and groups that could indicate collusive fraud.

The main concern in using the external rule data is that the rules information does not confirm fraud, it only gives indications and information to assist in investigations. Therefore even if accounts match against rules which typically indicate fraud, investigation must be performed and the application could be cleared if no hard evidence is found to the contrary. Experimenting with different selections and grouping of the data has exposed

a number of uses and cases that required further investigation. These included:
- Assisting in better understanding of rules that indicate fraudulent or non-fraud behaviour
- Identifying new fraud on book accounts by ranking/clustering via names, brokers & solicitors etc., postcodes/demographics, and bank account numbers
- Questions around policies and procedures used within the bank dealing with customers applying for several mortgages
- Identifying new targets for fraud models - for example rules which identify fraud in other banks

### 4.1.2 Visualization of Event-Rules for Fraud Detection

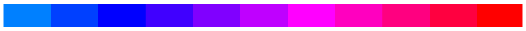

Figure 6. The colormap shows how the events, defined as system alerts in the form of rules, are colored: From blue to red the rules indicate more and more fraudulent behaviour.

Each line in the visualisations represents one account from the bank's database and aims to assist in fraud detection. The information on each account includes fixed data in the form of flags, such whether the account got a mortgage with the bank on the left hand side of the visualisations and time series events in the form of rules on the right hand side of the visualisations which indicate the possibility of fraud. The rules data are coloured according to the colour-map in Figure 6 with a rising fraudulent probability from blue to red. The fixed flags include whether accounts got on book, how they are performing and whether they have been found to be fraudulent post completion; see the colors at the beginning of each line in Figures 7 or 8 (Green for "Case on book" (obtained a mortgage with the bank) and Red for "Case allocated a fraud flag post obtaining a mortgage with the bank").

### 4.1.3 Findings

The first case was identified when visualizing and ranking the data by solicitors. Figure 7 shows a solicitor that at first sight was linked to a number of known fraud on book entries (18 red flags) and several other suspicious entries matching against rules post completion that typically indicate fraud. Further investigations revealed the solicitor had already been removed from the panel but visualizing the solicitor's business has instigated investigations of 14 cases for fraudulent behaviour. In Figure 7 this is can be seen in the entries without a red flag but with a red ending event.

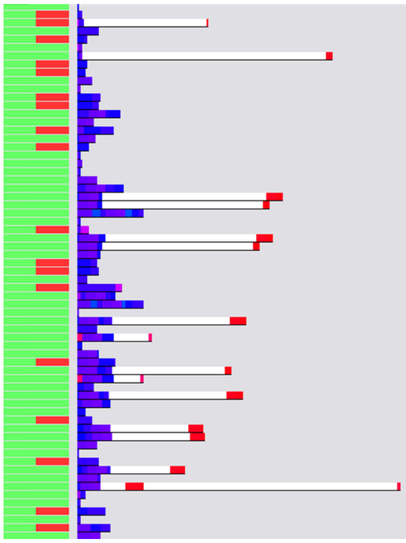

Figure 7. Figure is visualizing data for one solicitor with 18 known fraud on book cases with a red flag and 14 visually identified cases for fraudulent behaviour.

The next case shown in Figure 8 was identified when visualizing and ranking the data by bank account numbers. It exposed a number of cases where the same bank account number had been entered at application stage. Of the cases which had successfully completed (green flags); a proportion had already been identified as fraudulent post completion (red flags), importantly, the visualisation tool was able to flag a number of linked accounts. The Fraud team had previously flagged these as fraud after identifying income fraud collusion between these customers using the same bank account number. The group of individuals in question were all part of the same family and owned a property business together. The other 7 entries linked to this bank account number are presently being investigated and are likely to be assigned fraud flags.

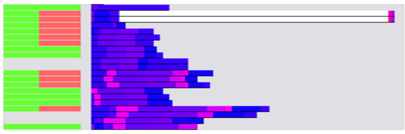

Figure 8. Figure is visualizing one bank account number used by several cases and customers. 10 cases out of the 17 on book have been flagged as fraudulent but the Fraud team were not aware of the other 7 using the same bank account number. These are presently being investigated.

The final case shown in the fraud detection application area was identified when visualizing the data ranked again via solicitors. At first sight in Figure 9 the solicitors business is all clear and performing well (no red or blue flags). But the matched time series rules data shows seven of the cases linked to this solicitor raised some suspicions. In Figure 9 they can be seen in the entries with a green flag and with a red ending event. Further investigations exposed that the solicitor was being monitored and the seven suspicious cases were split between two customers. All entries were performing well but the volume of mortgages and type of rules being fired raised suspicion and further investigations on these two individuals are being carried out.

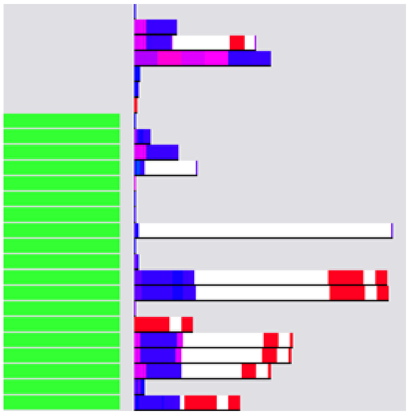

Figure 9. Visualized data for one solicitor for whom business is all clear and performing well (no red or blue flags) but several cases are matching against fraud rules post completion raising suspicions of possible fraudulent behaviour.

### 4.2 Sentiment Analysis in News Feeds

This second case study is about online news monitoring with respect to emotional debates about selected entities. In particular, we used our visualization technique to display event episodes, in which the absolute temporal reference plays an important role. Our data set was also used in Wanner et al.[26] where another visualization method for event data is presented. In this case study we aim to find the same events as they found there and more.

### 4.2.1 Data Set and the Visualization of Event Episodes

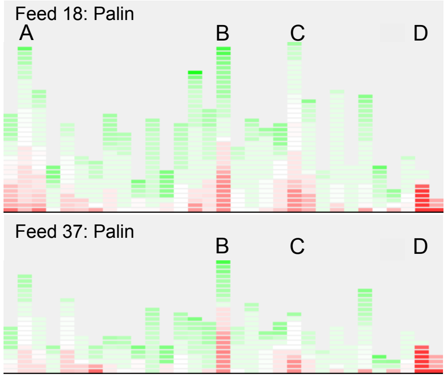

Figure 10. Sarah Palin mentioned in a negative connection in two feeds 18 and 37: the red items in stack (A) and (B) indicate that *she abused her power*, (C) further news regarding "abusing her power by charging the state when her children traveled with her", (D) After the election Palin bought a *too expensive wardrobe*. You can see in the cross-feed analysis between feeds 18 and 37 above that both reported very similar.

The data

> *was gathered from 50 different RSS news feeds that mainly dealt with the 2008 US presidential elections. The RSS feeds were retrieved every 30 minutes during a time interval of one month (10/09/2008 - 11/10/2008). For every news item in each feed we saved date, title and description, as well as the id of the feed. Next, noise was eliminated out of the title and description. With noise we refer to strings that do not carry any content" with respect to our sentiment annotation, "such as URLs or strings consisting of special characters. The concatenation of title and description was then considered to be the content of the news item. Finally, we filtered out those documents that contained none of the following signal words: "Obama", "McCain", "Biden", "Palin", "Democrat" and "Republican". More than 23000 news items contained at least one of the six strings.*[26]

Since we are interested in emotional debates, we enrich each event with a sentiment score. We therefore *"make use of a freely available list of words that evoke positive or negative associations.*[6] *We count the number of positive and negative words and evaluate the whole news item as rather positive if it contains in total more positive than negative words. Likewise, the item is evaluated as rather negative if it contains more negative than positive words. The absolute relation of positive against negative words normalized by the event's length, provides our sentiment score."*[26] Finally, for the visualization task of this case study we normalized the sentiment score to a score between 0 and 100, where 0 means very bad sentiment, 50 marks a neutral event and 100 denotes very positive news. It is important to know that the occurrence of a name of a presidential candiate only means that the person is in a negative context but this is not inevitably bad publicity for the respective person in terms of the campaign.[26]

Figure 11 shows a screenshot of one of the 50 monitored news feeds. Each horizontal black line represents the baseline of the news for the respective entity. In total we show six entities: Obama, McCain, Biden, Palin, the Democratic party and Republican party. Based on the first black line you can see all the news belonging to Obama posted by the feed with ID 37. Every news posting is represented through a red, white or green rectangle. All events of one day are sorted according to their sentiment score and arranged in a vertical stacked bar. In contrast to the previous case study, we do not aim at displaying event sequences with only relative temporal references, but rather event episodes with an absolute daily temporal reference. Each day is represented by one vertical bar of events, which enables us to do cross-entity and cross-feed comparisons since the temporal alignment is fixed. Furthermore, for better visibility of the proportions between positive and negative events, we sort the events according to their sentiment score within each day.

### 4.2.2 Findings

Already on the second day of our data collection many negative news postings are visible about *Sarah Palin* as shown in Figure 10. Almost all red marked articles deal with the topic whether she had *abused her power in Alaska* or not. The same topic reappeared on another day, when many negative news postings occurred about Sarah Palin. Cluster (B) of intense red shapes symbolizes this bad news coverage of her. Five days later, cluster (C) displays further negative news turning up: "A new ethics complaint has been filed against Sarah Palin, accusing the Alaska governor of abusing her power by charging the state when her children traveled with her". After the election some very negatively rated events stick out in cluster (D). These news deal with some critical notes about the *expensive wardrobe*, which was bought by Sarah Palin for her campaign, and her inappropriate use of language describing her critics. These event episodes were also found by Wanner et al.[26] The visual clutter and overlapping free cross feed analysis capability shown here, extends the functionality of their approach.

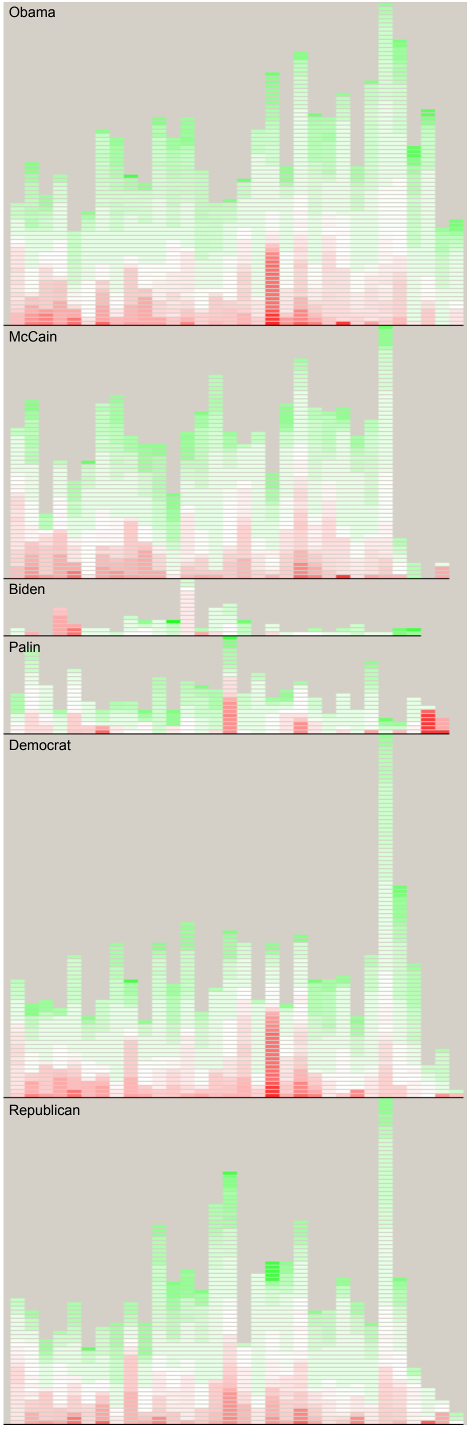

Figure 11. Cross-entity analysis showing more news about Palin than Biden and a high number of mostly positive postings on the election date November 4, 2008.

Cross-feed analysis in Figure 10 shows that both Feeds 18 and 37 reported very similar on the topic, which is due to the fact that they both used postings of the same news agency as the basis for their articles. Cross-entity analysis as shown in Figure 11 enables comparison of different entities. In this case, through interpretation of the two diagrams in the middle, it immediately becomes obvious that the Republican vice presidential candidate Palin was a lot more in the news than her Democratic counterpart Biden, whereas the total amount of news about each of the two parties in the lower two diagrams is comparable. Approximately one week before the US presidential election day we detected a high appearance of news which included Obama and Republican (see Figure 11, the intense red stacks). The sentiment scores of these postings were mainly negative and dealt generally with a plot to assassinate Barack Obama. A further remarkable event pattern is the extremely high number of mostly positive postings on the election date November 4, 2008 as seen for all entities in Figure 11. This is followed by a steep drop of news about the unsuccessful Republican presidential candidate McCain. Like in[26] we are able to detect meaningful event episodes within the data. Furthermore, we offer an analyst a possibility of cross feed and cross entity analysis offering a clutter and overlapping free visualization.

## 5. DISCUSSION

Time series visualizations heavily depend on the fact that the displayed data can be aggregated or are spread sufficiently in time so that no overlap occurs. However, for many practical applications neither of these properties hold since many events occur at the same time or long periods elapse without event activity. In such a case, time series visualizations typically degrade, which means that a lot of display space is wasted while still not all events can be displayed due to an overlap problem. Our proposed event data visualization tackles exactly these two shortcomings by rendering each atomic event and by abstracting or leaving out long temporal gaps in the representation. Thereby, our method has proven to be a flexible approach for finding significant events, event clusters and event patterns. The first case study, in particular, demonstrated the tool's capability to deal with event sequences, which are ordered but whose absolute temporal reference is irrelevant for the analysis. Based on real data from a bank's mortgage fraud database, we were able to find several event patterns, such as potential fraud cases of suspicious solicitors, a suspicious bank account shared by several fraudulent customers, a systematic mortgage application pattern of one customer and potential future risks on book. The second case study demonstrated the tool's capability to handle event episodes, which are time-stamped and in which the absolute reference in time is an important aspect. Using political RSS news feeds about the U.S. presidential election in 2008, we were able to identify significant events, such as a positive denial of an obvious scandal, event clusters such as feeds reporting very similar about one candidate, and event patterns like emotional debates.

## 6. CONCLUSIONS

In this paper we proposed a novel and flexible system for analyzing timed event data with advanced features such as similarity ordering, temporal gap alignment and stacking of co-occurring events. In particular, our system supports the analyst in the identification of significant events, search for event clusters and detection of event patterns, which was demonstrated on two characteristically different case studies. The first case study dealt with event sequences, where the ordering of events is more important than their absolute temporal references. By visualizing fraud detection events from a bank's mortgage department, we were able to find a number of event patterns. The second case study dealing with event episodes, in which the absolute temporal reference plays an important role, analyzed emotional debates of the U.S. presidential election in 2008 based on political RSS news feeds. In this data set, we were able to detect significant events, event clusters, and event patterns. Besides making the tool more user friendly and to improve the similarity sorting, we plan to extend our tool towards a matrix representation to correlate temporal intra-day patterns across several days. We believe that this will be of particular interest when analyzing stock trading patterns.

## REFERENCES

[1] Aigner, W. ; Miksch, S. ; Muller, W. ; Schumann, H. ; Tominski, C. : Visual methods for analyzing time-oriented data. In: *IEEE Transactions on Visualization and Computer Graphics* 14 (2008), Nr. 1, S. 47–60

[2] Ankerst, M. ; Keim, D. A. ; Kriegel, H.-P. : Recursive Pattern: A Technique for Visualizing Very Large Amounts of Data. In: *Proc. Visualization '95, Atlanta, GA*, 1995, S. 279–286

[3] Ankerst, M. ; Keim, D. A. ; Kriegel, H.-P. : Circle Segments: A Technique for Visually Exploring Large Multidimensional Data Sets. In: *Visualization '96, Hot Topic Session, San Francisco, CA*

[4] Bederson, B. B. ; Clamage, A. ; Czerwinski, M. P. ; Robertson, G. G.: Date-Lens: A fisheye calendar interface for PDAs. In: *ACM Trans. Comput.-Hum. Interact.* 11 (2004), Nr. 1, S. 90–119. `http://dx.doi.org/http://doi.acm.org/10.1145/972648.972652`. – DOI http://doi.acm.org/10.1145/972648.972652. – ISSN 1073–0516

[5] Bertini, E. ; Hertzog, P. ; Lalanne, D. : SpiralView: towards security policies assessment through visual correlation of network resources with evolution of alarms. In: *IEEE Symposium on Visual Analytics Science and Technology, 2007. VAST 2007*, 2007, S. 139–146

[6] Buvac, V. : *Internet General Inquirer.* 2008. – http://www.webuse.umd.edu:9090/ as retrieved on Nov. 14

[7] Chang, R. ; Ghoniem, M. ; Kosara, R. ; Ribarsky, W. ; Yang, J. ; Suma, E. ; Ziemkiewicz, C. ; Kern, D. ; Sudjianto, A. : WireVis: Visualization of categorical, time-varying data from financial transactions. In: *IEEE Symposium of Visual Analytics Science and Technology*, 2007, S. 155–162

[8] Galton, A. ; Augusto, J. C.: Two Approaches to Event Definition. In: *DEXA '02: Proceedings of the 13th International Conference on Database and Expert Systems Applications.* London, UK : Springer-Verlag, 2002. – ISBN 3–540–44126–3, S. 547–556

[9] Gatalsky, P. ; Andrienko, N. ; Andrienko, G. : Interactive analysis of event data using space-time cube. In: *International Conference on Information Visualisation* Bd. 8, 2004, S. 145–152

[10] Guralnik, V. ; Srivastava, J. : Event detection from time series data. In: *KDD '99: Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining.* New York, NY, USA : ACM, 1999. – ISBN 1–58113–143–7, S. 33–42

[11] Hao, M. ; Keim, D. ; Dayal, U. ; Schreck, T. : Importance-Driven Visualization Layouts for Large Time Series Data. In: *IEEE Symposium on Information Visualization (InfoVis 2005)*, 2005

[12] Hao, M. ; Keim, D. ; Dayal, U. ; Schreck, T. : Multi-Resolution Techniques for Visual Exploration of Large Time-Series Data. In: *Eurographics/IEEE-VGTC Symposium on Visualization, Norrkping, Sweden*

[13] Hochheiser, H. ; Shneiderman, B. : Dynamic query tools for time series data sets: timebox widgets for interactive exploration. In: *Information Visualization* 3 (2004), Nr. 1, S. 1–18

[14] Keim, D. A. ; Hao, M. C. ; Dayal, U. ; Hsu, M. : Pixel Bar Charts: A Visualization Technique for Very Large Multi-Attribute Data Sets. In: *Visualization, San Diego 2001, extended version in: Information Visualization Journal, Palgrave* 1 (2002), Nr. 2

[15] Keim, D. A. ; Nietzschmann, T. ; Schelwies, N. ; Schneidewind, J. ; Schreck, T. ; Ziegler, H. : FinDEx: A Spectral Visualization System for Analyzing Financial Time Series Data. In: *EuroVis 2006: Eurographics/IEEE-VGTC Symposium on Visualization, Lisbon, Portugal, 8-10 May*, 2006

[16] Kiernan, J. ; Terzi, E. : EventSummarizer: a tool for summarizing large event sequences. In: *EDBT '09: Proceedings of the 12th International Conference on Extending Database Technology*. New York, NY, USA : ACM, 2009. – ISBN 978–1–60558–422–5, S. 1136–1139

[17] Koutsofios, E. E. ; North, S. C. ; Truscott, R. ; Keim, D. A.: Visualizing large-scale telecommunication networks and services (case study). In: *VIS '99: Proceedings of the conference on Visualization '99*. Los Alamitos, CA, USA : IEEE Computer Society Press, 1999. – ISBN 0–7803–5897, S. 457–461

[18] Kumar, N. ; Lolla, N. ; Keogh, E. ; Lonardi, S. ; Ratanamahatana, C. A.: Time-series bitmaps: a practical visualization tool for working with large time series databases. In: *SIAM 2005 Data Mining Conference*, SIAM, 2005, S. 531–535

[19] Mannila, H. ; Toivonen, H. ; Inkeri Verkamo, A. : Discovery of Frequent Episodes in Event Sequences. In: *Data Min. Knowl. Discov.* 1 (1997), Nr. 3, S. 259–289. `http://dx.doi.org/http://dx.doi.org/10.1023/A:1009748302351`. – DOI http://dx.doi.org/10.1023/A:1009748302351. – ISSN 1384–5810

[20] Mansmann, F. ; Keim, D. A. ; North, S. C. ; Rexroad, B. ; Sheleheda, D. : Visual Analysis of Network Traffic for Resource Planning, Interactive Monitoring, and Interpretation of Security Threats. In: *IEEE Transactions on Visualization and Computer Graphics* 13 (2007), Nr. 6. `http://infovis.uni-konstanz.de/papers/2007/InfoVis2007NetworkSecurity.pdf`

[21] McLachlan, P. ; Munzner, T. ; Koutsofios, E. ; North, S. : LiveRAC: interactive visual exploration of system management time-series data. In: *CHI '08: Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*. New York, NY, USA : ACM, 2008. – ISBN 978–1–60558–011–1, S. 1483–1492

[22] Phan, D. ; Gerth, J. ; Lee, M. ; Paepcke, A. ; Winograd, T. : Visual Analysis of Network Flow Data with Timelines and Event Plots. In: *VizSEC 2007*, Springer, 2008

[23] Rosling, H. : *Gapminder*. – Available from http://www.gapminder.org/. Accessed on Mar. 30, 2010

[24] Suntinger, M. ; Obweger, H. ; Schiefer, J. ; Groeller, M. E.: Event Tunnel: Exploring Event-Driven Business Processes. In: *IEEE Computer Graphics and Applications* 28 (2008), S. 46–55. `http://dx.doi.org/http://doi.ieeecomputersociety.org/10.1109/MCG.2008.97`. – DOI http://doi.ieeecomputersociety.org/10.1109/MCG.2008.97. – ISSN 0272–1716

[25] Van Wijk, J. J. ; Van Selow, E. R.: Cluster and Calendar Based Visualization of Time Series Data. In: *Proceedings of the IEEE Symposium on Information Visualization*, IEEE Computer Society, 1999, S. 4–9

[26] Wanner, F. ; Rohrdantz, C. ; Mansmann, F. ; Oelke, D. ; Keim, D. A.: Visual sentiment analysis of rss news feeds featuring the us presidential election in 2008. In: *In Workshop on Visual Interfaces to the Social and the Semantic Web (VISSW)* Citeseer, 2009

[27] Weber, M. ; Alexa, M. ; Muller, W. : Visualizing Time-Series on Spirals. In: *Information Visualization, IEEE Symposium on* 0 (2001), S. 7. `http://dx.doi.org/http://doi.ieeecomputersociety.org/10.1109/INFVIS.2001.963273`. – DOI http://doi.ieeecomputersociety.org/10.1109/INFVIS.2001.963273. – ISSN 1522–404X