Identification of Suspicious, Unknown Event Patterns in an Event Cloud

Alexander Widder[1], Rainer v. Ammon[1], Philippe Schaeffer[2], Christian Wolff[3]

[1]Centrum für Informations-Technologie Transfer GmbH, D-93051 Regensburg, Germany, alexander_widder@gmx.de, rainer.ammon@citt-online.com,
[2]TÜV Rheinland Secure iT GmbH, D-51105 Cologne, Germany, philippe.schaeffer@de.tuv.com,
[3]Media Computing, University of Regensburg, D-93040 Regensburg, Germany, christian.wolff@sprachlit.uni-regensburg.de

ABSTRACT

This paper describes an approach to detect unknown event patterns. In this context, an event is not only something that happens, but also something that can be analysed. This task is processed by a Complex Event Processing (CEP) engine. CEP is an emerging technology for detecting known patterns of events and correlating them to complex events in real-time. In order to reach the goal of finding unknown patterns, several known detection algorithms are discussed. In our work, we focus on discriminant analysis used for recognizing unknown patterns for the use case of credit card transactions and the fraud problem connected with this kind of payment. It is necessary to develop new methods of fraud detection and prevention because of the negative impacts for vendors and customers caused by credit card fraudsters at present. At the same time we would like to make provisions for the more sophisticated fraud methods that will occur in the future.

Categories and Subject Descriptors

I.5.2 [**Pattern Recognition**]: Design Methodology – *Pattern analysis*; H.4.2 [**Information Systems Applications**]: Types of Systems – *Decision support (e.g., MIS)*

General Terms

Algorithms, Design, Experimentation, Security, Languages

Keywords

Complex Event Processing, Event Driven Architecture, Service Oriented Computing, Business Activity Monitoring, Discriminant Analysis, Fraud Detection

1. INTRODUCTION

With headwords like *ubiquitous* and *pervasive computing* or *ambient intelligence*, a new computing paradigm has been established in the last ten years. Networked computing technology now penetrates almost all aspects of our life and working environment. Most papers that are written on these topics discuss them from a mere technical point of view [27]. This is not surprising, as tremendous advances have been achieved in the last years, e.g. sophisticated sensors. But there are also initiatives which consider these themes in the direction of organisations and users. This area is called *ambient business intelligence* which can be seen as the next generation of the widespread business intelligence (BI) systems. Data analysis methods in traditional BI systems rely on predefined cubes which do not reflect the real-time-business [11]. One central concern of next generation BI systems is the ability to deal with real-time data that originates e.g. from operations, message queues or web clicks [24]. This is fundamental for realizing predictive business systems. This is an environment where users can access data they need in real-time, analyze it and predict possible problems and trends with the aim of optimising enterprise decisions [6]. One part of predictive business and a solution for delivering information in real-time is complex event processing (CEP). CEP platforms scan low level events, e.g. on the network level like SNMP traps or database commits. Such events, occur in the global event cloud [20, pp.28-29] of an enterprise, without any business relevant semantics. They generate complex, business level events in real-time if a predefined event pattern matches to an occurring combination of events, e.g. for credit card fraud or intrusion detection and prevention.

A CEP engine will be able to react to specific events in real-time. Event processing is endorsed by software analysts and vendors as one of the emerging styles of programming and software architecture (e.g. the Event Driven Architecture (EDA) [26]). Today many applications require event-based monitoring capabilities ranging from digital data streaming systems, continuous query systems, system monitoring and management tools to event-driven workflow engines. There is a wide and growing interest in event processing techniques both for extending the capabilities and for improving the performance and ease of use of such EDA systems. While industry solutions are evolving, the scientific community also deals with fundamental issues behind the modelling, the representation, the usability and the optimization of event processing [26]. Event processing is still a young discipline. Officially it has been founded and established as a discipline with a community around it in March 2006 [12]. Event processing systems are widely used in enterprise integration applications, ranging from time-critical systems, agile process integration systems, managements of services and processes, delivery of information services, and awareness of business situations. There is a range of event processing middleware capabilities, including publish-subscribe services, which have been incorporated into standards such as CORBA and JMS, and into commercial systems, mediation services such as event transformation, aggregation, split and composition, and event pattern detection [26].

2. EVENT CLOUD, TYPES OF EVENTS, EVENT PATTERNS

In the global event cloud of an organization many kinds of events exist. According to [20, p.88] an event is a record of an activity in

DEBS '07, June 20–22, 2007, Toronto, Canada.

a system and may be related to other events. It has the following aspects:

- *Form*: These are the formal attributes of an event, such as timestamp, place or originator.
- *Significance*: It is the activity, which signifies the event.
- *Relativity*: This describes the relationship to with other events. An event can be related to other events by time, causality, and aggregation. It has the same relations as the signified activity of the event [20, p.88].

Since 2006 a discussion on the proper definition of the event concept has started inside the CEP community. According to a very wide interpretation "an event is simply anything what happens". Other members of the community suggest a more restrictive definition: "an event is a notable activity that happens" [7]. In comparison with transactions, which can change permanently, events are static. If a transaction changes, a new event of the new state will be created [16, p.6]. Events can be high level business events like "depositing funds into a bank account" or low level events like acknowledging the arrival of a TCP-packet. By the use of CEP-engines, low-level events can be aggregated to high level events. This can be achieved with known event patterns.

2.1 Known Event Patterns

Known event patterns can be derived from heuristics e.g. from a specific business activity monitoring view (BAM-view). The event patterns are implemented using event pattern languages (EPL). An EPL must have the following properties:

- *Power of expression*: It must provide relational operations to describe the relationships between events.
- *Notational simplicity*: It must have a simple notation in order to write patterns succinctly.
- *Precise semantics*: It must provide a mathematically precise concept of matching.
- *Scalable pattern matching*: It must have an efficient pattern matcher in order to be able to handle large amounts of events in real-time [20, p.146].

Examples of EPL's are RAPIDE-EPL, STRAW-EPL, StreamSQL and there are still ongoing research efforts [7, 14, 9].
An event-pattern written with STRAW-EPL looks like this:

Element	**Declarations**
Variables	Node N1, Node N2, Data D, Bit B, Time T, Time T1, Time T2
Event Types	Send(Data D, Bit B, Time T), Receive(Data D, Bit B, Time T), Ack (Data D, Bit B, Time T), RecAck (Data D, Bit B, Time T)
Rational operators	-> (causes)
Pattern	Send (D, B, T1) -> Receive (D, B) -> Ack (B) -> RecAck (B, T2)
Context test	T2 – T1 < 10 sec
Action	**create** Warning (N1, N2, T1, T2)

This pattern describes a TCP data transmission. If the time between the send-event and the recack-event is more then 10 seconds, a warning-event will be created. This warning-event is a complex event with the parameters node N1, node N2, time T1 and time T2 [20, p.117]. Examples for the use of known event patterns are discussed in chapter 3.

2.2 Unknown Event Patterns

In contrast to known event patterns, unknown event patterns can not be derived from heuristics based on an existing BAM view. They did not exist in the past respectively they have not been recognized so far. An unknown pattern could be found with the help of event processing agents by analysing the event cloud of an organisation and using specific algorithms to detect it. This approach is discussed in detail in chapter 4.

2.3 Risks of the of Patterns Approach

There are also risks connected with the patterns approach. On the one hand, a pattern can be too specific, so it does not match situations where a reaction is necessary. The reason for not reacting is that events defined in the pattern are not occurring. On the other hand an event pattern can be too general and fires too often. The result is that the pattern produces alerts in situations, where it is not necessary [8, p.3]. Therefore it is important to find the right combination of relevant events for fulfilling the approach of VIRT (Valuable Information at the Right Time) which is concerned with filtering, compacting and delivering information to the right place at the right time [15]. Moreover event patterns must be continuously improved and updated.
Another peril of using patterns lies in the "acclimatization factor". If whatever complex process relies on the accurate and automated processing of events, the user or business case gets used to the fact that any occurring events are automatically handled correctly. But especially in the case of unknown event patterns, i.e. a combination of events that has not been considered so far and therefore no appropriate handling of these events has been defined, the automatic handling of events may not lead to the desired results, e.g. important events may not be taken care of. But since the user or business case is used to relying on the automated process the wrong or incomplete results may not be noticed.

3. TECHNIQUES TO DETECT KNOWN EVENT PATTERNS

There are a high number of domains, which are interested in finding known event patterns on the base of occurring events at runtime, e.g. health care, military or insurance companies to name just a few. The following use cases are focused on the detection of known event patterns in the banking domain.

3.1 Use Case: Fraud Detection

The types of fraud in the banking domain are versatile. They reach from phishing over cross-site scripting to credit card fraud. Some vendors are focusing on developing anti-fraud systems [16, p.9]. The event pattern shown in fig. 1 is used for fraud detection in a billing process. If a bill submitted for payment on the billing system, has an invalid customer or billing information, an email to the billing manager will be sent. Fig.2 shows a more complex example for a known event pattern.
If one day after a submitted transaction the address or the password is changed and a loss of the card is reported while invalid transactions on this account have occurred in the last ten days, then the defined actions will be executed. In this case, the reactions predefined in the pattern are
1. Putting activities on referral by the account system,
2. Investigating the transaction by the personal event manager and
3. Suspending the transaction.

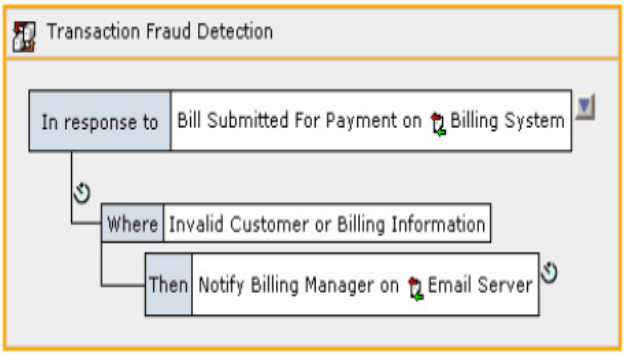

Figure 1: Pattern for fraud detection in a billing process [2]

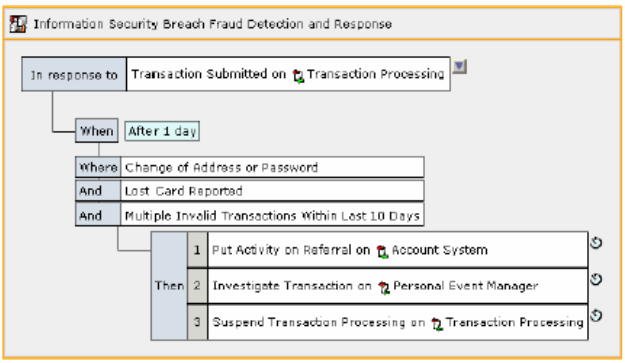

Figure 2: Pattern for fraud detection in the case of transaction processing [2]

Fig. 3 shows the evolution of multiple invalid transactions in the last ten days before 03/15/2005.

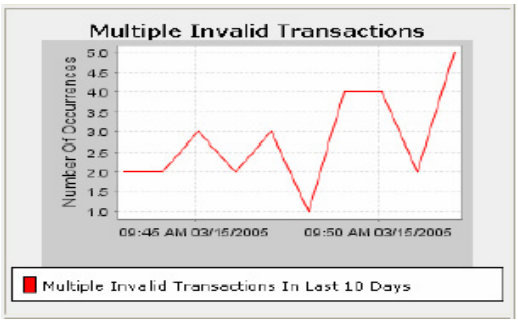

Figure 3: Evolution of invalid transactions [2]

Fig. 4 documents the trend of password changing in the last 10 days before 04/06/2005.

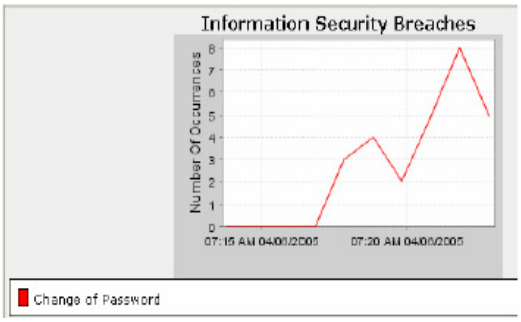

Figure 4:Trend of password changes[2]

CEP platforms typically deliver many more patterns for fraud detection (see [2] for details). According to a recent Federal Trade Commission (FTC) report, its consumer sentinel complaint database received more than 635.000 consumer fraud and identity theft complaints in the year 2004. Furthermore, a 2003 study by the US-based Identity Theft Resource Center, a non-profit organization focusing on identity theft, estimated the business community losses between $40.000 and $92.000 per name in fraudulent costs [2]. These total costs consist of:

- *Direct fraud costs*: Costs caused by successful fraudsters.
- *Costs of manual order review*: These are the costs of checking orders manually by humans.
- *Costs of reviewing tools*: These are the costs of tools which check orders automatically.
- *Costs of rejecting orders*: These are lost turnovers caused by falsely evaluated orders [8, p.10].

In the next paragraph we discuss well-known pattern matching algorithms from application domains like information retrieval or artificial intelligence. Our goal is to find out whether such algorithms may be used for event pattern detection.

3.2 Algorithms Used for Pattern Matching and Recognition

The following methods and algorithms are used for detecting known event patterns. They are also candidates currently discussed as solutions for detecting *unknown* event patterns [6]:

- *Deterministic approaches*: They describe processes, which are stringently causal. E.g. event A causes event B and event B is leading to event C and no other variant is possible [10].
- *Probabilistic approaches*: In contrast to deterministic approaches, probabilistic approaches are not stringently causal. E.g. event A causes event B with a specific probability and event C with a different probability [1].
- *Cluster operations*: These methods are creating groups (clusters) of objects out of a basic set of objects on the basis of specific criteria. Many kinds of cluster algorithms exist, e.g. the k-nearest Neighbour algorithm (KNN) [25].
- *Discriminant analysis*: This method checks the quality of existing group divisions by means of classification methods, which are based on discriminant functions [21].
- *Fuzzy set theory*: This approach extends the classical binary truth function in set theory by introducing degrees of membership of an object to a set in the interval from 0 to 1 [13].
- *Bayesian Belief Networks*: This method generates inferences based on unsure information. It is a network graph whose nodes are states and the edges between the nodes describe the dependences between a pair of states [18].
- *Dempster-Shafer method*: This method is also known as the evidence-theory. It combines information from different sources to a total conclusion [28].
- *Hidden Markov models (HMM)*: This method describes two random processes, one of which is hidden. With the help of the probability distribution of the known process the probability distribution of the hidden process is determined [23].

These algorithms are only a sample of the methods used for pattern recognition. For our goal to detect unknown event patterns, we will use combinations of these algorithms. This is also men-

tioned in [5, p.6] for the domain of intrusion detection (see also chap. 6). Our fist step will be to investigate discriminant analysis.

3.3 Known Fraud Scenarios and Methods Used for Handling Fraud Management

According to [8, p.2], a survey which interviewed 150 UK online retailers about their experiences with fraud and how they defend themselves against crime, fraudsters have a wide range of tricks. The most popular method of them is to use stolen credit cards. In this context, they try multiple identity details with the same credit card number until they find a combination which is able to pass the security system. This way, they often test a stolen card by ordering small volumes of low-value products. After a test order is successful, they will continue to use the stolen card until the limit is reached. Moreover, thieves often use card holders' real addresses for placing an order and afterwards change the delivery address to an address where they can pick up the goods. This can be achieved by contacting the specific call centre before the order is delivered. Furthermore, retailers often report problems with foreign orders, especially orders which originate from Africa. According to the survey, retailers are most afraid of fraudsters which use sophisticated methods just as the above mentioned identity theft [8, p.13]. In order to meet these threats, most of the retailers increased their investments in fraud management between 10% and 100% over the year [8, p.12]. In this context they use fraud management methods which are shown in fig. 5.

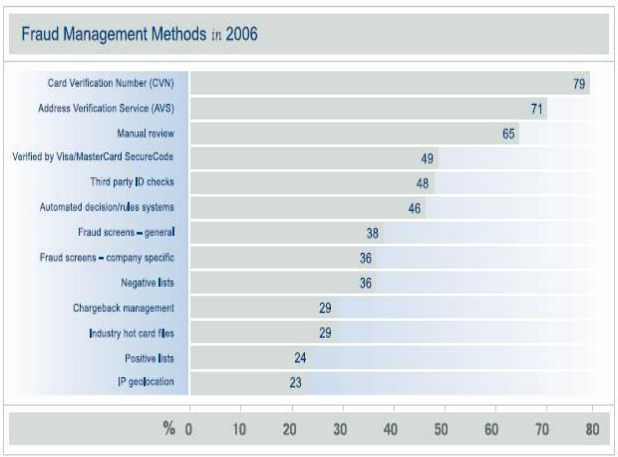

Figure 5: Fraud management methods used by UK online retailers [8, p.9]

Most of the consulted retailers (79%) use Card Verification Number (CVN) in the year 2006. The purpose of CVN is to verify that the person placing an order has the credit card in its possession. To accomplish this, the CVN is requested during an online purchase process. Address Verification Service (AVS) is also widely used by the online retailers (71%). In addition, manual review by humans is a popular method to enhance the credit card security (65%). Fig. 5 also shows that many retailers use a combination of two or more fraud management methods, e.g. manual review after automatic detection tools as well as CVN.

3.4 Shortcomings of Fraud Detection Systems

In spite of widespread implementations of AVS and CVN, these systems offer some weaknesses: For example, AVS displays a problem if the address of the card holder is not up to date. In this case, the address will be flagged as invalid. The result is that AVS has a significant rate of false-positives. On the other hand the verification number of CVN can be obtained by fraudsters [8, p.8]. Furthermore, the "Hot Card File" which contains information about stolen or copied cards is not always a reliable source of data and can be out of date for several days. This is because the file depends on card owners to recognize that a fraud has happened and report it [8, p.10]. In addition, sophisticated fraudsters know the length of time a card is registered in the file or try tactics to remove it from the file e.g. by flooding the file with false card numbers until the targeted card number drops out of the list.
In general, according to [2], traditional anti-fraud systems narrowly focus on transactional activities such as opening a new credit card account or changing a password. But these events often happen in disparate systems at different times and so they may not be detected by current anti-fraud detection technology [2]. Moreover, because of more sophisticated fraud methods, the known kinds of fraud patterns change permanently: Thus, they are not detectable, but leave traces in the form of unknown fraud event combinations. One approach to find unknown event patterns will be discussed in the following chapter.

4. AN APPROACH FOR DETECTING UNKNOWN EVENT PATTERNS USING DISCRIMINANT ANALYSIS

We suggest the scenario shown in fig. 6 as a possible approach to recognize unknown event patterns:

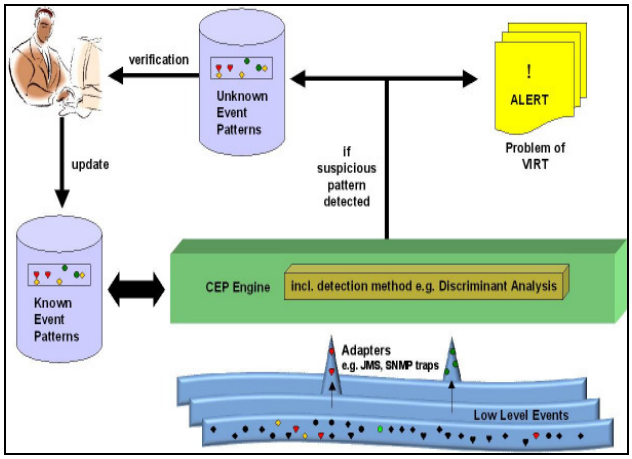

Figure 6: An *Event Processing Agent* detects unknown event patterns in a CEP engine

Event processing adapters map low level events occurring in the global event cloud or respectively event streams from their specific structure into a standardized format (e.g. the Common Base Event (CBE) [17]) which can be interpreted by an Event Processing Agent (EPA) inside the CEP engine. If an EPA detects an unknown pattern which seems to be a fraud attempt or in other words a suspicious combination of events as determined by using e.g. discriminant analysis, it will react with a predefined action, like sending an alert to a target system. But the best solution would be to start a preventing action which stops the fraud attempt before it is finished. Furthermore, the suspicious pattern will be saved in a database of unknown event patterns. The next step is a verification process by a responsible person who classi-

fies the new pattern as harmless or as a dangerous fraud pattern. This operation is followed by an update of the database which stores the known event patterns. The next paragraphs exemplify the application of discriminant analysis as an appropriate approach for identifying unknown event patterns.

4.1 The Principle of Discriminant Analysis

Discriminant analysis is a multivariate statistical method. Such methods analyse multidimensional data in order to help finding decisions in economical applications or to discover relationships between certain kinds of data. Discriminant analysis in particular consists of the following functions:

- It checks the quality of membership of objects in predefined groups of objects in order to discover the optimal discrimination between the groups.
- It allocates a new occurring object into one of the existing groups of objects [21, p.300].

The process to determine the optimal group a new object belongs to can be described as follows: First, the parameters relevant for distinguishing the groups must be defined. On the basis of these variables, discriminant functions that separate the groups will be calculated. In this step the multi-group case or the two-group case can be applied. In the two-group case, only one discriminant function exists. The form of this function depends on the number of the variables differentiating the groups, e.g. if two appropriate variables exist, the function will have the form:

$$Y = V1*X1+V2*X2$$

X1 and X2 are the values of the specific parameters of the new object. V1 and V2 are the coefficients of the discriminant function. These coefficients can be computed by including the values of the parameters of the existing objects in a linear system of equations. The result is Y, which is the discriminant value of the new occurring object. The next step is to compare the computed discriminant value of the new object with the so called critical discriminant value. The critical discriminant value of a discriminant function is the midpoint of the average discriminant values of the two groups [29]. If the computed discriminant value of the new object is greater than the critical discriminant value, than the new object will be allocated to the group of objects with the greater discriminant values (and vice versa). Another way to define the membership of an object to a group is to use Fisher's linear discriminant function [21, p.318] which has a critical discriminant value of null. In this way, the group membership of an object depends on the algebraic sign of the discriminant value of the specific object.
In the multi-group case, more discriminant functions exist. The first function compares group A with the summation of the other groups. If the new object does not belong to group A, the second discriminant function compares group B with the remaining groups without A. This algorithm is finished when the optimal group for the new object is found. So the maximal count of discriminant functions is: number of groups – 1. This classification process is described in more detail in [21, pp.300-333]. In order to accomplish a discriminant analysis, the following preconditions should be fulfilled:

- The number of parameters should be greater than the number of groups.
- The range of the sample should be double the amount of the number of the parameters.
- The basis data should be normally distributed.
- An object must not belong to more than one group.
- The values of the variables must be metrically scalable.

Typical use cases for discriminant analyses are:

- On the basis of balance key figures a bank decides whether a company is creditworthy or not.
- On the basis of patient data, diseases can be recognized earlier.
- On the basis of aptitude tests, the success of beginners in a certain job could be predicted.

A new use case for discriminant analysis will be an approach to classify events in order to recognize unknown event patterns.

4.2 Fraud Detection Based on CEP and Discriminant Analysis

The first step in order to detect suspicious patterns by means of discriminant analysis is to define different use case specific groups of events. These groups can be created via cluster analysis algorithms on the basis of the existing events in the event cloud. Of course, a great number of different groups can be defined to classify the event-groups more accurately. But to simplify the process, this approach is restricted to two groups of events: Events which are indicators for credit card fraud and events which are not relevant in this sense. After defining the groups, the second step is to compute a discriminant function, which differentiates the groups. The parameters of the discriminant function are two or more metric event-attributes which have to be appropriate for identifying the kind of events and to show differences between the groups for the specific use case. If the needed kind of event does not have significant metric attributes, the CEP adapter has to convert string attributes into metric forms. The third step is to determine the critical discriminant value, which will be compared with the discriminant value of an event. By using this method, a new occurring event can be allocated to one of the – in this case – two groups. The following example shows this approach: A person inserts a credit card in an automatic teller machine (ATM) followed by changing the password and the withdrawal of money within a predefined period of time. This operation creates events just like SNMP traps [17]. Some (metric) attributes of these events will be included in the predefined discriminant function. The resulting discriminant value will be compared with the critical discriminant value and afterwards allocated to a group of events, in this case to the group of suspicious events. The described process is executed by the CEP engine in real-time. To meet the high performance demand of realizing this process in real-time, grid computing can be used respectively splitting the event stream and distributing the load to several computers. In the meantime, there are first products like Tibco Business Events [16, p.30] or Kaskad [16, p.27] which support grid computing. There are still some questions which must be answered for the approach based on discriminant analysis:

- Which types of events are created by a credit card transaction and which are important to detect credit card fraud?
- Which attributes of the event types are relevant to differenciate the groups of events for the use case credit card fraud detection?
- Does the CEP adapter have to change string attributes to metric values?

4.3 Discriminant Analysis inside a Reference Architecture

Some vendors like IBM, Oracle, or Tibco provide CEP reference architectures. Fig. 7 contains the reference model of [3], which is designed for "Predictive Business". The main process of this reference model has the following task: It predicts the business impact of occurring events or situations by processing real-time events together with historical data stored in DB's. The occurring events originate from heterogeneous event sources. The results are outputs (e.g. alerts) displayed in an user interface.

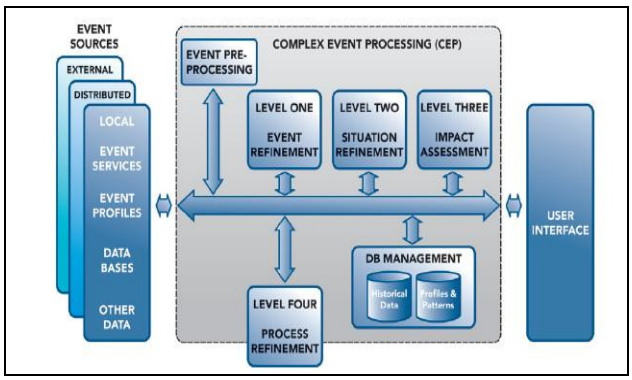

Figure 7: CEP reference model [3]

To achieve the goal of this reference model, there are several levels of event processing that events have to pass through:

- *Level 0*: Event Pre-processing (normalization, transformation, data cleansing on raw data)
- *Level 1*: Event Refinement (event identification, event tracking, event pre-selection, selection of "events of interest")
- *Level 2*: Situation Refinement (situation identification based on relationships between event-objects and historical data such as signatures, profiles, and other DB info)
- *Level 3*: Impact Assessment (estimating the impact of complex events on the organization and business processes)
- *Level 4*: Process Refinement (adaptive, dynamic adjustment of processes based on the overall processing architecture, including turning event sources on/off, bringing new DB's online, changing algorithms, adjusting parameters, filters, etc. both automated and with human interaction)
- *Database management*: (storing features and patterns extracted from historical data and other databases e.g. databases of known IP addresses of internet fraudsters) [3]

To enhance such a reference model, discriminant analysis as detection algorithm can be integrated in level 2 (Situation Refinement, shown in fig. 8). Here it can be used as an internal function for distributing events to different groups (see ch. 4.2).

But in this context, the computation of the discriminant function and the critical discriminant value is not only based on existing events but also on parameter values of historical data (e.g. events of known fraud attempts) in order to allocate new real-time events more exactly to a predefined group of events. So the groups of events are more accurate and can represent an unknown pattern of events itself. This approach results in a decrease of false positives and contributes to solve the problem of VIRT [15]. If an unknown pattern is detected and validated, the next step is to update the historical data with the new fraud attempt. Afterwards the discriminant function will be computed again and und updated in real-time. Thus, the discriminant function becomes more and more accurate.

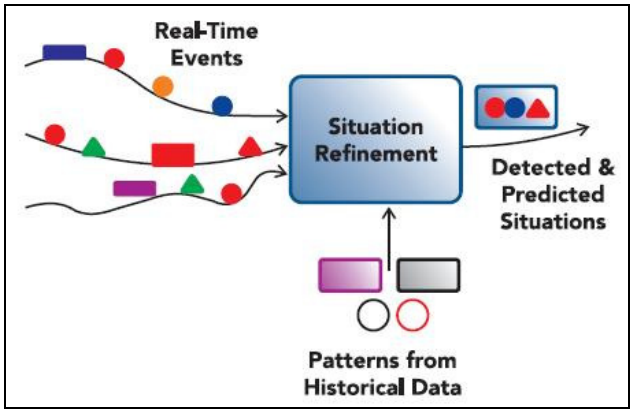

Figure 8: Process of Situation Refinement [3]

5. RELATED WORK

In consideration of known and unknown patterns for the application areas intrusion detection and fraud detection and the recognition of anomalies and intrusions, there is an approach at [19]. That method was developed before 2001 and is based on data mining approaches with the particular background of real-time processing problems. The focus of this work was to develop cost-sensitive models for the distribution of "features" to detect and process patterns on more areas and secondly their optimal distribution in the infrastructure architecture. But now, with CEP technology the basis is developed to handle events in real-time, e.g. via grid computing technology, as mentioned in paragraph 4.2.

A further research, according to [22], examines the accuracy of probabilistic methods in the field of naive Bayes text classification. To be more detailed, the classification accuracy of the mulit-variate Bernoulli model and the multinomial model are compared. In this context, an event is declared as the occurrence of a specific word inside a text document. The experiments are based on different data sets just as Yahoo Science, Newsgroups, WebKB and the Industry Sector. The research results showed that on the one hand the mulit-variate Bernoulli model performs better with small vocabulary sizes but on the other hand the multinomial model usually is better suited for larger vocabulary sizes. That is only an example for applying probabilistic or fuzzy methods from the discipline of information retrieval which were already developed in the seventies and the following years of the 20th century and which could be researched and perhaps adapted for detecting unknown event patterns.

The algorithms, as well as the mathematical and heuristic techniques from fields such as statistics, artificial intelligence, operations research, digital signal processing, pattern recognition, decision theory etc. discussed here are also presented in [4]. In that paper a new generation of Intrusion Detection Systems is discussed and in the meantime this is going to be one of the main applications of CEP. But until today the authors don't know how and whether these sophisticated techniques are really already applied in concrete products respectively projects. This is also discussed in the work of [5, p.12]. He found out that the existing commercial products are not focused on the problem of detecting unknown event patterns. His own approach is based on data-mining and not on CEP. Furthermore, one of his results was that

the combination of different methods [5, p.6] (e.g. Bayes network of probability of each intrinsic attribute, Matching against non-self bit-vectors, Hidden Markov Model etc.[5, p.41]) for detecting all types of intrusion attacks leads to the recognition that this highly specialized approach only works for the domain of intrusion detection but can not applied for other domains like online fraud detection [5, p.11]. Finally, the first trial of combining methods should be implemented in the years after 2004 and experimental and empirical results are expected these days.

6. CONCLUSIONS

This paper proposes discriminant analysis as a possible approach in order to classify events into specific groups of events where a group can represent an unknown pattern itself. But to determine the meaning of every specific group of events as well as defining the used event parameters for developing the discriminant function depends on the types of occurring events and on the concerning use-case. At the moment, we implement an experimental environment based on a CEP engine in order to test our discriminant analysis approach. After finishing this step, we are going to examine other detection algorithm candidates (see ch. 3.2) as well as combinations of these algorithms. Our goal is to evaluate their abilities to detect unknown event patterns and to be able to compare the different methods.

7. REFERENCES

[1] Alon, N., Joel, H., and Spencer, J. *The Probabilistic Method*. Wiley InterScience, New York, 2000.

[2] AptSoft Corporation. CEP Solution. http://www.aptsoft.com, accessed 2006-12-22.

[3] Bass, T. Fraud Detection and Event Processing for Predictive Business. http://www.tibco.com/resources/mk/ fraud_detection_in_cep_wp.pdf, accessed 2007-01-31.

[4] Bass, T. Intrusion Detection Systems and Multisensor Data Fusion. CACM 43(4) (2000), 99-105 (online: http://www.silkroad.com/papers/ pdf/acm-p99-bass.pdf, accessed 2007-03-07).

[5] Brugger, T. Data Mining Methods for Network Intrusion Detection. University of California at Davis, 2004.

[6] CEP Interest Group. http://tech.groups.yahoo.com/group/ CEP-Interest, Blog entry Thu Oct 5 2006 4:01 pm form leondong1982, accessed 2006-12-09.

[7] Complexevents.com. http://complexevents.com/?cat=15, accessed 2006-12-06.

[8] CyberSource. Third Annual UK Online Fraud Report. http://www.cybersource.co.uk/resources/ fraud_report_2007, accessed 2007-02-07.

[9] Demers, A., Gehrke, J., Panda, B., Riedewald, M., Sharma, V., and White, W. Cayuga: A General Purpose Event Monitoring System. In Proceedings of the third Biennial Conference on Innovative Data Systems Research, Asilomar, 2007.

[10] Earman, J. *A Primer on Determinism*. Springer-Verlag, Dordrecht, 1986.

[11] Fernandes, L. Mainstream BI: A dashboard on every desktop?. http://www.it-director.com/enterprise/ content.php?cid=9035, accessed 2006-12-06.

[12] First Event Processing Symposium. http://complexevents.com/?p=150, accessed 2006-12-29.

[13] Gottwald, S. *A Treatise on Many-Valued Logics*. Research Studies Press LTD, Baldock, Hertfordshire, 2001.

[14] Gyllstrom, D., Diao, Y., Stahlberg, P., Chae, H., Anderson, G., and Wu, E. SASE: Complex Event Processing over Streams. In Proceedings of the third Biennial Conference on Innovative Data Systems Research, Asilomar, 2007.

[15] Hayes-Roth, F. Model-based communication networks and VIRT: Orders of magnitude better for information superiority. In Proceedings of the Military Communications Conference, Washington, 2006.

[16] Howard, P. The market for event processing is growing and at some point, it will explode. http://www.bloor-research.com/ research/ research_report/802/ event_processing.html, accessed 2006-12-22.

[17] IBM Autonomic Computing, Best practices for problem determination mean faster problem resolution. http://www-03.ibm.com/autonomic/pd/pdf/ AC_PD_whitepaper_V4.pdf, accessed 2007-02-10.

[18] Jensen, F. *Bayesian Networks and Decision Graphs*. Springer-Verlag, New York, 2001.

[19] Lee, W., Stolfo, S., Chan, P., Eskin, E., Fan ,W., Miller, M., Hershkop, S., and Zhang, J. Real Time Data Mining-based Intrusion Detection. In Proceedings of the second DARPA Information Survivability Conference and Exposition, Anaheim, 2000, pp. 85-100.

[20] Luckham, D. *The power of events.* Addison Wesley, San Francisco, New York, 2002.

[21] Mardia, K.V., Kent, J. T., and Bibby, J. M. *Multivariate Analysis*. Academic Press, San Diego, San Francisco, New York, Boston, London, Sidney, Tokyo, 1979.

[22] McCallum, A., and Kamal, N. A Comparison of Event Models for Naive Bayes Text Classification. AAAI-98 Workshop on "Learning for Text Categorization". http://www.kamalnigam.com/papers/multinomial-aaaiws98.pdf, accessed 2007-02-15.

[23] Rabiner, L. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. In Proceedings of the IEEE 77(2), 1989, pp. 257-286.

[24] Raden, N. Ambient Business Intelligence: Pervasive technology to surround and inform. http://www.hiredbrains.com/Ambient%20Business%20%Intelligence.pdf, accessed 2006-12-06.

[25] Romesburg, C. *Cluster Analysis for Researchers*. Lulu Press, Morrisville, 2004.

[26] Schloss Dagstuhl. Event Processing Seminar. http://www.dagstuhl.de/en/program/calendar/semhp/ ?semid=32202, accessed 2006-12-28.

[27] Schoder, D. Ambient Business. http://www.wim.uni-koeln.de/Ambient-Business.430.0.html, accessed 2006-12-21.

[28] Shafer, G. *A Mathematical Theory of Evidence*. Princeton University Press, Princeton, 1976.

[29] Wallrafen, J. Genetically Optimized Neuronal Network Classifiers for Bankruptcy Prediction. In Proceedings of the 29th annual Hawaii International Conference on System Sciences, Maui, 1996.