Data-Driven Fraud Detection Using Detectlets

Conan C. Albrecht
Chad O. Albrecht*

Fraud detection is becoming increasingly important to managers of organizations, to internal and external auditors, and to regulators (Apostolou and Crumbley, 2005). Recent large frauds in many countries of the world and the Sarbanes-Oxley Act in the United States stress the importance of early detection of fraud. Financial statement frauds have weakened investor confidence in corporate financial statements (Castellano and Melancon, 2002), led to a decrease in market capitalization (Palmrose, et al, 2004), and contributed to five of the ten largest bankruptcies in U.S. history. These four include WorldCom at $104 billion, Enron at $65 billion, Conseco, Inc. at $61 billion, Pacific Gas and Electric at $36 billion, and Refco, Inc. at $33 billion. In addition, the impact of recent investment scams were most visible in the alleged $50 billion Madoff fraud. Internationally, financial statement misstatements such as Parmalat (Italy), Harris Scarfe and HIH (Australia), SKGlobal (Korea), YGX (China), Livedoor Co. (Japan), Royal Ahold (Netherlands), and Vivendi (France) indicate that fraud is a worldwide problem.

Because a $1 fraud against an organization reduces net income by $1 and because organizations usually have profit margins of 10 to 20 percent, additional revenue of 5 to 10 times the amount of the fraud must usually be generated to restore net income to its pre-fraud level. For example, a major automobile manufacturing company had a $436 million fraud a few years ago (Albrecht, 2001a). At the time, the company's profit margin was just under 10 percent, meaning that additional revenues of approximately $4.36 billion had to be generated to bring net income to what it would have been without

* The authors are, respectively, Assistant Professor at Brigham Young University and Visiting Assistant Professor at Utah State University.

the fraud. Assuming automobiles sell for an average price of $20,000 each, the company had to make and sell 218,000 additional automobiles to restore net income to its pre-fraud amount. If this fraud had been proactively detected earlier, the fraud loss would have been much smaller and the effect on the firm much less severe.

Despite actions taken in many countries designed to increase the discovery of fraud, many frauds are still discovered much too late. In its most recent report, the Association of Certified Fraud Examiners found that auditors find only 30 percent of reported frauds, and most of these frauds are found only when they became egregious enough to cause significant damage (ACFE, 2004). And even these numbers are subject to some skepticism because many frauds are never discovered nor reported (Lohr, 1997). Frauds progress exponentially as perpetrators become greedy and gain confidence in their schemes. A common saying in fraud circles is that a small fraud is only a large fraud caught early.

The Data-Driven Fraud Detection Method

Albrecht and Albrecht (2001b) found that the data-driven method of fraud detection was effective at discovering occupational fraud. The data-driven method is a six-step process that effectively represents a hypothesis-testing approach. It is depicted in Figure 1. First, the detective team understands the business by studying corporate policies and procedures, financial statements, by interviewing employees, and by observation. Next, the team brainstorms the possible fraud symptoms that might exist within the context being studied. These schemes may come from a variety of sources, including internal auditors, security personnel, line employees, and management. For each scheme, the resulting symptoms are identified. Since fraud is rarely witnessed directly, detection usually focuses around the symptoms, or red flags, that it causes (Albrecht and Romney, 1986). This process usually produces around 50 schemes with 3-5 symptoms each. The team then uses technology, such as database

queries, to discover if the symptoms exist in the business data. After analysis, the team follows up with traditional investigative techniques. (See Appendix A, Figure 1).

The data-driven approach is different from the traditional approach to fraud detection in many ways. Table 1 lists several of the most significant differences.

Table 1: Traditional vs. Data-Driven Fraud Detection

Traditional Fraud Detection	***Data-Driven Fraud Detection***
Reactive in nature: Fraud detection begins when an indicator is seen in the regular audit process. This usually means the fraud has become large enough that it cannot be missed.	Proactive in nature: Fraud detection begins as part of a repeatable process. No indication of fraud is present at the time the detection begins.
Typically analyzes only samples of the available data, potentially missing the fraudulent transactions.	Analyzes entire population of available data. Since fraud often occurs in very few transactions, this approach ensures all transactions are analyzed.
Requires knowledge of less-advanced software like MS Excel or specialized audit sampling software.	Requires advanced knowledge of fraud, symptoms, database queries, and scripting to analyze full populations.

I. Failings of the Data-Driven Approach

Reed and Pence (2005) wrote that many auditors are looking for a "magic bullet" to identify troubled companies at the beginning of an audit. In the author's experience of consulting for and teaching thousands of auditors and fraud investigators, most investigators want a wizard-based approach. In short, many want to purchase an install disk containing fraud detection software that starts a wizard. The first page of this wizard asks simply "Would you like to detect fraud today?". When the investigator clicks "Yes", the wizard automatically finds all appropriate databases, compares data against standard models, and determines the exact fraud occurring in a business. Most are simply not

willing to learn the advanced techniques (both fraud-related and computer-based) required to find fraud in data sets.

While the data-driven method provided some direction toward this magic bullet and despite research supporting the strategic method's effectiveness (Beasley and Jenkins, 2003), it has been criticized in industry because of its "heavyweight" approach to fraud detection. Indeed, the proponents of the strategic approach have conceded that the strategic method is costly and time-intensive (Albrecht, et. al. 2006). It requires advanced skills in computer and data analysis fields. In particular, the paper identifies five reasons the strategic approach has not seen wide adoption: lack of time, fraud tools, technical knowledge, scheme knowledge, and symptom knowledge.

Lack of Time

Auditors and most other professionals are increasingly stretched to complete tasks with lower cost and shorter time frames. In this environment, auditors are often pushed to their limits of performance and attention. Herbert Simon has been often quoted as saying that human attention would become the most significant scarce resource as the pace of business increased (McQuaid, et. al., 2000). With the addition of the Sarbanes-Oxley ruling in the United States (and comparable standards in other countries), auditors find their time increasingly limited and their required tasks multiplying.

Brainstorming 50 fraud schemes often results in 250-500 total symptoms that have to be researched and followed up on. While the initial brainstorming takes only a few hours, following up on the symptoms is a significant task. Since each search may take several hours, this adds significant time and resource requirements to audits that are already stretched for time. Braun (2000) showed that auditors under greater time pressures were less sensitive to fraud cues, and therefore, less likely to detect

fraud. Applying Braun's research to the strategic method, it could be argued that in practice the strategic method might even decrease effectiveness.

Lack of Fraud Methodology and Supporting Tools

During most of the last century, auditors were not required to find fraud (Winters and Sullivan, 1994 and Epstein and Geiger, 1994). Only recently have audit standards (SAS 82 and SAS 99) increased auditor requirements related to fraud. Therefore, most of today's methods are centered around the discovery of routine errors rather than intentional frauds. In addition, many auditors do not realize that fraud symptoms are very different from regular accounting anomalies (Albrecht, et. Al, 2006). Accounting anomalies, such as control weaknesses, occur in regular and predictable patterns. Anomalies are not intentional; they are just errors in the system. Fraud symptoms, on the other hand, are intentionally hidden; perpetrators always work around the controls (or the fraud wouldn't be possible), making auditing of controls ineffective for fraud discovery. While anomalies are routine and predictable, fraud symptoms often occur in very few, seemingly random transactions. Zimbelman (1997) found that SAS 82, an audit standard requiring auditors to pay more attention to fraud risks, made mild progress but failed to change the nature of routine audits. Marczewski and Akers (2005) found that CPAs don't anticipate that the recent SAS 99 ruling will substantially improve audit effectiveness.

Sampling methods, heavily used by auditors, is effective at generalizing routine anomalies to the population of transactions in an organization, but its effectiveness is negligible in finding. Frauds are by default not representative of the general population. The use of sampling is becoming less important as auditors become more skilled on full-population programs like ACL. However, sampling methods are still significantly used in routine audits. In addition, many other reasons keep fraud from being identified.

The two primary tools used by auditors, ACL and IDEA, are based in the error-finding tradition. They provide significant support for traditional analysis routines, such as sampling, gap detection, and sequencing, but they lack advanced fraud detection routines. Auditors who use these applications for real fraud detection typically write their own programs in the scripting languages of the two tools.

Lack of Technical Knowledge

The search for symptoms in the strategic method requires auditors to connect to many different types of databases, including transactional and relational databases, to retrieve support data. Auditors must be able to understand different types of databases, schema, table relationships, foreign keys and data formats to effectively query for information. Once information is queried, it is usually stored in a data warehouse. The information must be analyzed using a variety of routines, including digital analysis techniques such as Benford's Law (Nigrini, 1999), time series analysis, positive and negative comparisons, stratification and combination, etc.

While some auditors have experience with smaller databases like Microsoft Access, the complex analysis routines required are topics more suited to an Information Systems (IS) graduate rather than an Accounting graduate. Yet these skills, in addition to accounting skills, are crucial to the success of the strategic method. Auditors with significant expertise in data analysis and fraud detection can use these applications effectively for fraud detection. Most of the auditors doing the detail audit work—recent graduates and other inexperienced accountants—do not have this type of expertise. While IS members of a fraud team can provide some support for these types of routines, there are simply not enough IS auditors to "go around". Some have called for auditors and computer forensics professionals to work together to bring the needed skills to detection teams (Smith, 2005).

Lack of Scheme Knowledge

While an increasing number of schools seem to be offering an introductory fraud class, many auditors have little experience with different fraud schemes. Since the standard fraud tools, ACL from ACL Services Ltd. and IDEA from CaseWare, provide no support for specific schemes, auditors must rely on their own resources to know how to use the applications for fraud detection. The author has received numerous emails from investigators trying to find books or web sites giving popular schemes and their indicators; while some basic books do exist, they are not detailed enough and do not tie investigation to today's electronic investigative techniques.

In addition, Butler, Ward, and Zimbelman (2000) found that auditors fail to understand the basic terms used in the professional literature related to fraud. When auditors do find information related to schemes and indicators, many do not have the training to understand them. This lack of foundation knowledge and background is illustrated in the following example on bid rigging.

Most auditors know that bid rigging is the illegal alteration of bids during the bidding process. However, very few auditors have experience with specific bid rigging schemes. As an example of a detailed scheme, consider a case one of the authors analyzed the data on. Bids were requested for a public works project for a European government. One way for bidders to collaborate is to designate a winner before bids are turned in. This bidder can overbid because he or she knows that the designated losers' bids will be even higher. Since he or she will win the bid, this bidder is tasked with preparing *all* of the bids (the designated losers don't want to waste time on a project they won't win). The bidder first prepares the winning bid in a spreadsheet program by specifying a charge for each of the line items on the bid in column A. The bidder calculates the second and third (losing) bids in columns B and C by multiplying column A cells by an exact percentage like 103.2 percent and 106.8 percent. The resulting numbers looked real to the human eye.

The scheme presented in the previous paragraph is the type of detailed knowledge auditors must gain to correctly identify schemes in different industries. It requires an understanding of the bidding process-how three bidders might collude, the motivations and relationships of the people involved, and how fraudulent bids are created. The scheme is underscored by a perpetrator who becomes lazy and complacent in his or her scheme.

Lack of Symptom Knowledge

In addition to specific scheme knowledge, auditors must know how the symptoms of different schemes show up in data. This requires the auditor moving to a new level of understanding: a synthesis of Information Systems knowledge with fraud scheme knowledge. Pincus (1989) found that the availability of extensive symptoms generated by checklists actually reduces the auditor's ability to use the information effectively. Hackenbrack (1992) and Waller and Zimbelman (2000) found that increasing numbers of cues correlated with less extreme and less accurate risk assessments. These studies show that without detailed scheme and symptom knowledge, additional indicators simply provide noise that confuse many auditors.

In our bid rigging example, the symptoms are not trivial to find. Given thousands of public works projects each year, the auditor must follow a more analytical approach to find the fraud. A fraud examiner should first separate the thousands of projects into separate tables—a table for the line items on each project. For each of these tables, the examiner must calculate a new column giving the difference (in percentage) between the amounts for each line item between two bids. Finally, the examiner should calculate the standard deviation of these differences, giving a single number showing how the line items are different across the line items of two bids. Most auditors do not have the experience with fraud to know that bids with line item standard deviations close to zero are likely symptoms of the fraud scheme described above.

II. Detectlets: A Potential Solution

The previous section detailed factors that keep auditors and investigators from discovering fraud. Table 2 summarizes these problems and the solutions required to address them.

Table 2: Why Fraud Is Not Detected

Problem	***Potential Solution***
Auditors do not have enough time to do comprehensive, detailed fraud investigations.	The solution must be as automated as possible and allow quick investigation. The use of computer algorithms and wizards can help to alleviate the time requirements in investigation.
Auditors do not understand basic fraud investigation methodology and supporting tool use.	The solution must embed the detection methodology to help walk auditors through the correct process.
Auditors lack technical knowledge, such as database access, query languages, and scripting for automation.	The solution must contain prewritten data query calls and scripts for each type of fraud searched for. This must be dynamic to allow use in different database schemas and adaptable to different uses.
Auditors have not been trained in scheme and symptom identification. Since symptoms can be matched to multiple schemes, and since schemes have multiple symptoms, auditors can get lost in the detail.	The solution must automatically highlight potential symptoms and help the auditor match symptoms to schemes. When multiple symptoms are found, the solution must help auditors pinpoint potential schemes and follow-up activities.

While the above limitations apply to most auditors, there are some exceptional individuals who have the background and experience to find fraud. These individuals have training in 1) auditing methods, 2) information systems and database topics, and 3) fraud scheme and symptom experience. They have experience in the fraud schemes used in different domains, such as the medical insurance, mining, and power industries. These individuals are skilled at scripting and data analysis, and they can effectively query and analyze large amounts of data. In short, they are capable of performing the steps of the data-driven approach with efficiency and precision.

The detectlet solution is a way to harness the expertise of these individuals so others (who may not have been trained in fraud detection) can successfully perform data-driven fraud detection. Instead of only providing the raw tools for analysis, detectlets embed domain knowledge and fraud detection routines to guide an auditor through the search for indicators. Detectlets provide a step-by-step approach using the familiar wizard paradigm. They ask the auditor questions and allow easy movement through the process via previous and next buttons. Detectlets first guide the user in collecting and identifying the required data for the analysis. Example data is provided to help the user understand how data should be structured and formatted. Once data are collected and validated, the detectlet runs the analysis to generate results. Since the routine knowledge is embedded in each detectlet, the auditor needs only to understand the principles being applied—he or she is freed from programming, complex statistical routines, and detailed fraud symptom knowledge. When it is finished, the detectlet displays the results and guides the user in interpreting them correctly.

In short, detectlets are a step towards the wizard-based approach that many auditors and investigators want. The architecture provides the methodology, background information, queries and analysis scripts to support each routine, and training required for regular auditors and investigators to conduct powerful searches.

Technically, detectlets are built on three technologies: Picalo, eXtensible Markup Language (XML), and Python. Picalo--described in section IV of this paper--is an open source fraud detection toolkit that specializes in retrieval of data from corporate servers, statistical analysis, outlier detection, time trend analysis, full-text matching and searching, and grouping and summarization of data sets. Detectlets use the foundation libraries in Picalo to perform their fraud detection routines. XML is a widely used data format that uses tags to describe data; detectlets use XML to describe the user interface wizard that is presented to the user. Finally, Python is a popular open source scripting language that is

widely known for its readability and ease of development. In detectlets, Python is the glue that specifies the processing order of Picalo routines and underlying logic to be used.

A Detectlet Illustration

The following example shows how a detectlet might codify the bid rigging scheme and symptom analysis presented earlier. Suppose a skilled individual recognized this scheme and codified its detection into a detectlet. He or she submits this detectlet as part of a library of other, related detectlets to the worldwide, open source repository. Sometime later, another auditor goes to the detectlet repository, downloads it, and installs it on his or her machine. (See Appendix A, Figure 2).

After connecting the program to the database, the auditor runs the bid rigging detectlet. Figure 2 shows the first screen, which teaches the auditor the basic scheme it will look for. It provides background information that allows the auditor to intelligently use the detectlet. This screen allows the auditor to remain at the conceptual level without being distracted by detailed technical information. The next few pages of the detectlet, shown in Figure 3, walk the auditor through the selection of input data. The detectlet makes example data available to further instruct the auditor on the tables, columns, and relationships required for this analysis. (See Appendix A, Figure 3).

After selection of input data, the detectlet wizard runs the analysis. The auditor does not need to be concerned with the details and complexity of the analysis. Internally (described for purposes of this paper), the detectlet first stratifies the data set by project, effectively producing 3,000 smaller tables (one for each bid). For each of the smaller tables, the detectlet calculates the mean and standard deviation of the differences between the line items in the winning bid and the other bids. It produces a results table containing entries for each project, sorted by the projects with the lowest standard deviation between line items from bidders.

Finally, the detectlet wizard displays the results table and an "Interpreting These Results" popup window, shown in Figure 4. A standard deviation of zero (0) indicates that all line items from one bidder are exact percentages above the line items from another bidder. Low standard deviations indicate that many line items are close percentages of one another. (See Appendix A, Figure 4).

Detectlets solve the problems identified earlier because they are fast and simple to run and because they encode the methodology, tool knowledge, and scheme and symptom knowledge. They allow the few auditors who have the skills required for computer-aided fraud detection to support the many auditors who may not have these skills.

III. The Picalo Platform

While detectlets are the basis of this approach, they require a support application to store and access data, provide foundation routines, and offer the wizard interface. This platform is the Picalo open source application, shown in Figure 5. The foundation (Level 1) routines provide the raw materials that detectlets use to analyze data. The platform provides the basis for a larger expert system that might enable automated fraud detection (described later). (See Appendix A, Figure 5).

Currently, the platform runs on Windows, Macintosh, and Linux, and it is programmed in an open way so detectlet creators can extend its core functionality and can add additional detectlets. While it contains only a few detectlets now, user contributions over time will make it a rich repository of fraud detection schemes and detection detectlets. As an open source application, it is free for firms, governments, and individuals. Its open source license protects the intellectual property the community contributes to it.

The Picalo platform provides a development environment within which auditors can create detectlets. The process of creating detectlets is called *detection engineering*, and those who create

detectlets are *detection engineers*. Since fraud has different symptoms across different situations, this process allows custom detectlets to be written for each setting. It allows detection engineers in one area to modify detectlets used in another area—enabling the community to "stand on each others' shoulders" in building this repository. If auditors and information systems professionals are willing to contribute, we'll soon have a repository of thousands of detectlets for many different domains and classes of fraud.

By allowing detection engineers to sell detectlets for profit or publish detectlets for free, the platform provides a marketplace and incentive for companies and individuals to get involved and contribute high quality libraries of detectlets. A rating system on the Picalo web site allows users to share feedback about detectlet quality and ease of use. The web site will also support user comments on individual detectlet libraries to allow discussion of schemes and indicators. As the project progresses, libraries like "detectlets for the discovery of medical insurance fraud", "detectlets for bid rigging in government contracts", "detectlets to discover revenue and expense fraud", and "detectlets that find overstated assets in mining" may be published in the repository.

IV. Automated Fraud Detection

One of the possibilities of this approach to fraud detection is the potential for automated fraud discovery by combining thousands of detectlets with artificial intelligence engines like Bayesian and neural networks, called Level 3 routines in the Picalo architecture. Some who are new to fraud detection naively want a discovery system that connects to databases, understands schema and table relationships for any domain and organization, and runs thousands of tests to automatically discover fraud with the click of a single button (Reed, 2005). This "detective-in-a-box" concept may not be realizable today, but detectlets may provide the foundation for such a system.

Two primary barriers exist for automated discovery. First, the individual detection routines must conform to some standard and be programmatically accessible so they can be combined together into larger programs. The detectlet structure is engineered toward this goal and should provide a solid foundation for overcoming this barrier. For example, the standard structure ensures that every detectlet can be accessed via a programming interface (rather than the wizard). In addition, detectlets always input and output Picalo tables, allowing detectlets to be chained together for higher-level functionality.

Second, the significant differences in databases, field names and types, relationships, and business processes used by organizations (even within a single industry) make full automation and accuracy very difficult. Work is underway to infer database relationships and schema to match against detectlets, but more detectlets are needed before these programs can be tested.

V. GOVERNMENT INVOLVEMENT

The detectlet methodology has been presented at many venues. While individuals around the world have downloaded and installed the Picalo software, two primary government entities are basing large fraud detection systems on the architecture: North Carolina in the United States and India's Ministry of Health. Both of these systems have wide scope and significant implementation; case study data will be collected as they are rolled out in the next few years. Some preliminary successes are described in the paragraphs below.

In North Carolina, the Office of the State Auditor is building a sophisticated, real-time detection system to be used throughout systems within the state. The group has hired four detectlet engineers to create scripts and detectlets to find many types of fraud common to its public universities, department of motor vehicles, health care, and other government bodies. The system is built on top of Picalo and hosted in the Python open source language.

A test version of the system was created in 2007 to find invalid use of social security numbers, the national id number in the United States. The system compared social security numbers used by

State University employees and motor vehicle registrations with the social security death index published by the federal government. The system found several employees using invalid numbers and thousands of drivers registered with numbers of recently deceased people (Daily Tar Heel, 2007). While the search is relatively simplistic (a simple matching of two databases), it was a proof-of-concept for the methodology. The larger production system will be finished sometime in late 2008.

The Ministry of Health and Family Welfare (MoHFW) in India will be the most significant test to date. The MoHFW is India's government health body at the national level; it is developing a new MIS system to track all health-related procurement, projects, and medicines used throughout the country. Using funding from the World Bank, detectlets are being engineered, written, and included to perform continuous monitoring of transactions moving through the system. The World Bank has provided support and training, and the system is currently in development by the Government of India. This large-scale implementation will provide a national-level case study of the effectiveness of the detectlet methodology.

VI. Conclusion

Picalo is currently at version 4.33. The Level 1 routines, graphical user interface, and Detectlet Wizard are complete, and the Picalo application are freely available at http://www.picalo.org/. The initial detectlet repository is online and ready to accept submissions from detectlet engineers. There is a significant amount of research needed by academics and PhD students in this architecture, particularly in the appropriate design of detectlets, schema matching to different detectlets, and expert system and artificial intelligence designs for Level 3 routines.

In the next few years, the worldwide repository of detectlets should fill up with submissions from different industries. This repository will represent an unprecedented, freely-available collection of all

types of fraud, including background information, scheme and symptom description, detection algorithm and script, and results interpretation. Most importantly, detectlets will bring this information to the day-to-day auditor in a user-friendly, wizard-based solution. While some may feel a repository with this information will inform perpetrators of potential schemes, it is believed that more transparency and information, not less, provides the most significant benefit.

The architecture has already proved valuable at representing employee and consumer frauds within the detectlet framework. However, whether management fraud (where management officials intentionally deceive external auditors and shareholders) can be represented within the detectlet framework remains to be seen. Most of the large recent frauds, such as HealthSouth, Enron, and WorldCom, were management frauds. Further research is needed to determine how these frauds map into the detectlet framework.

Additional research is also needed to determine how detectlets can be plugged into different database schemes found within various organizations. Academic work has been done in the past few decades on schema translation, and this research can be referenced to make detectlets more dynamic and generalized.

Finally, research is needed to determine how different detectlets can be combined in an expert system that performs automated fraud detection. While this "auditor in a box" concept may never be fully recognizable, detectlets may prove to be the foundation of a system that provides some level of automation.

Appendix A

Figure 1: The Data-Driven Approach to Fraud Detection

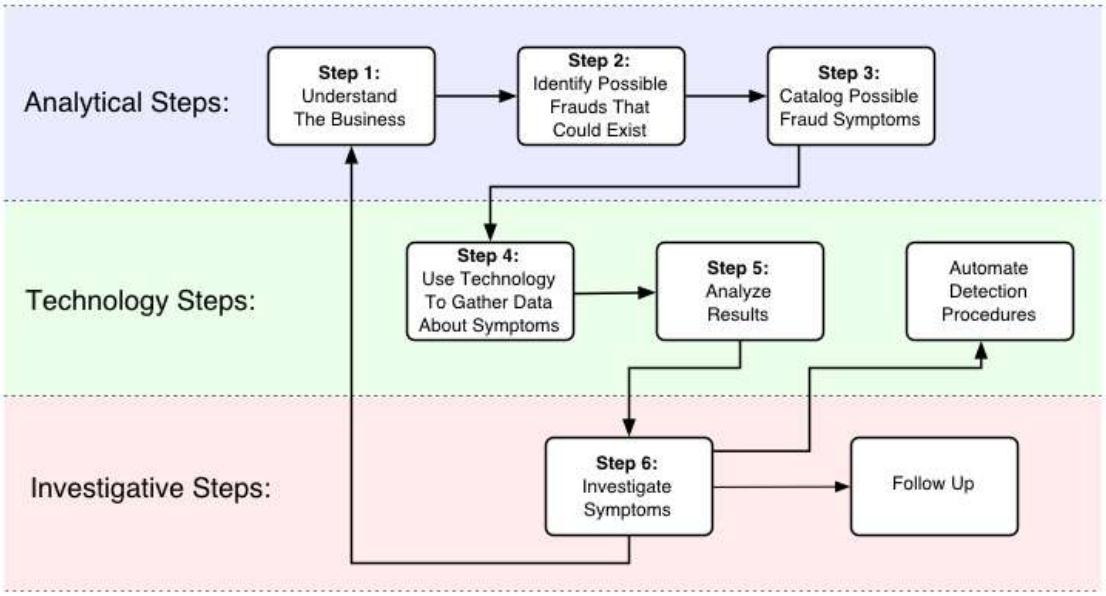


Figure 2: Detectlet Introduction Screen (Windows Interface)

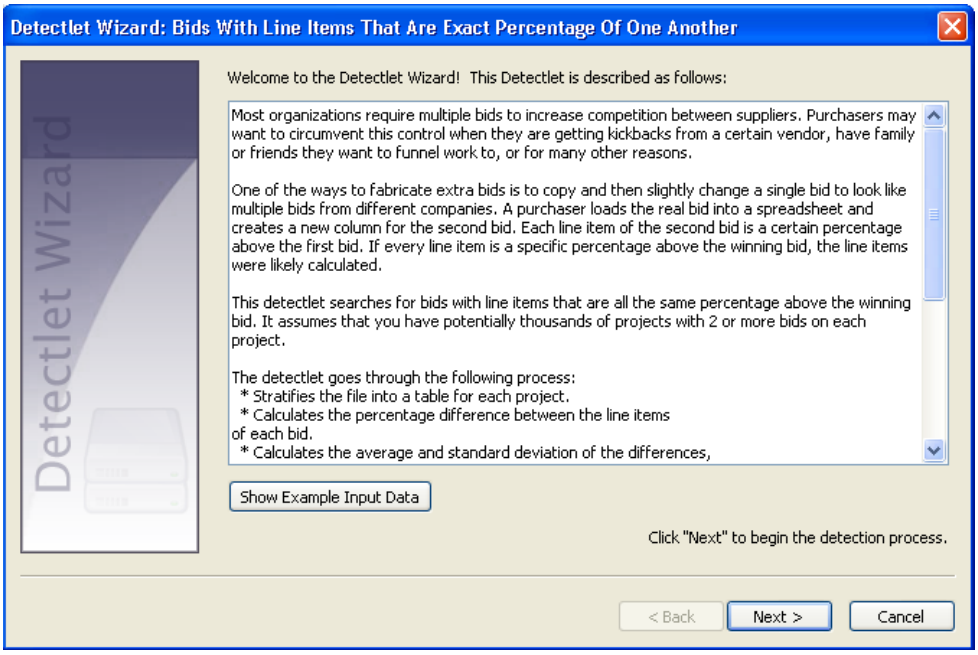


Figure 3: Selection of Input Data

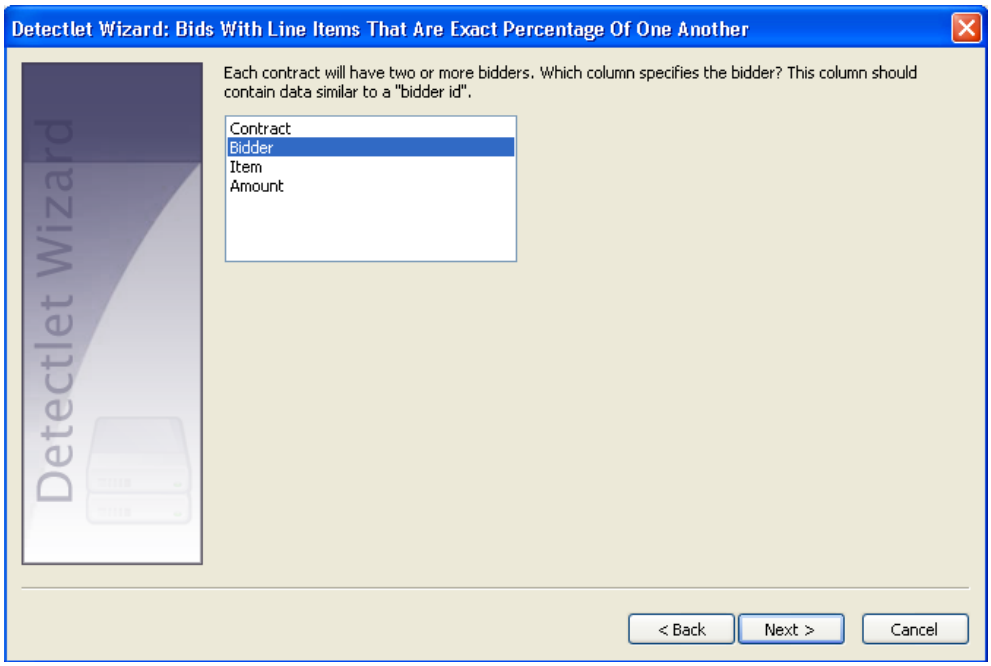


Figure 4: Detectlet Results (Mac OS X Interface)

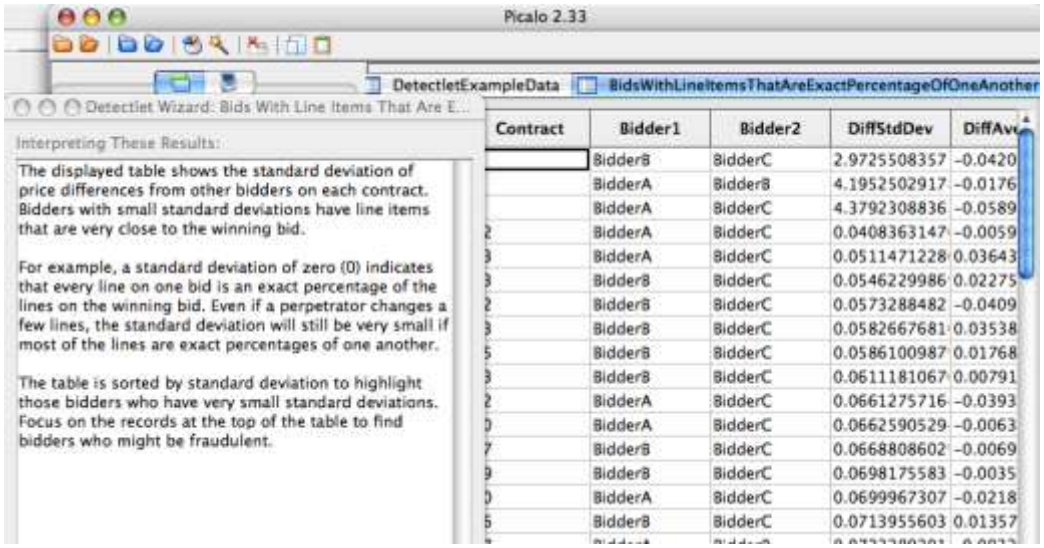
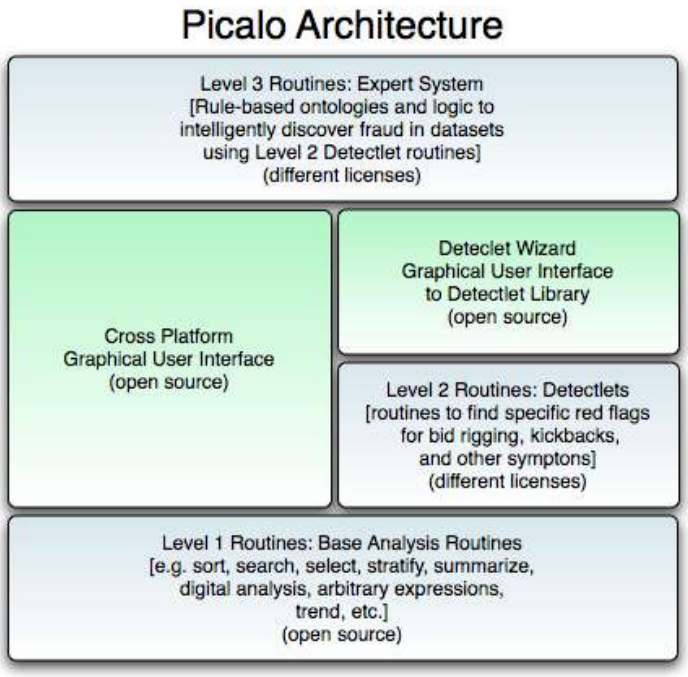


Figure 5: Picalo Architecture



References

AICPA. 2002. SAS No. 99: Consideration of Fraud in a Financial Statement Audit Summary, *AICPA*.

ACFE, 2004. 2004 Report to the Nation on Occupational Fraud and Abuse, Association of Certified Fraud Examiners. http://www.cfenet.com/pdfs/2004RttN.pdf

Albrecht, C. C., W. S. Albrecht, et. al. 2001a. "Can Auditors Detect Fraud: A Review of the Research Evidence." *The Journal of Forensic Accounting I*: (January-June) 1-12.

Albrecht, C. C., W. S. Albrecht, et al. 2001b. "Conducting a Pro-Active Fraud Audit: A Case Study." *The Journal of Forensic Accounting II*: (June-December) 203-218.

Albrecht, W. S, Albrecht, C. C., and Albrecht, C. O. 2006. Fraud Examination, Thompson.

Albrecht, W. S. and Romney, M. B. 1989. Red-Flagging Management Fraud: A Validation. *Advances in Accounting*, volume 3, pages 323-333.

Apostolou, N. and Crumbley, D. L. 2005. Fraud Surveys: Lessons for Forensic Accountants. *Journal of Forensic Accounting,* vol 6, pages 103-118.

Beasley, M. S. and Jenkins, J . G. 2003. The Relation of Information Technology and Financial Statement Fraud. *Journal of Forensic Accounting*, vol 4, pages 217-232.

Braun, R. L., 2000. The effect of time pressure on auditor attention to qualitative aspects of misstatements indicative of potential fraudulent financial reporting. *Accounting, Organizations, and Society*. 25: 3, pages 243-259.

Butler, S., Ward, B. and Zimbelman, M. 2000. The expectation gap: Auditors' and investors' perceptions of auditors' fraud detection responsibilities. *Working Paper*. Brigham Young University.

Castellano, J. G. and Melancon, B. C. 2002. Letters to Members, *AICPA*. http://www.aicpa.org/info/letter_02_01.htm, Accessed on September 9, 2004.

Daily Tar Heel. 2007. Report on social security number fraud. University of North Carolina at Chapel Hill, April 26.

Epstein, M. and Geiger, M. 1994. Investor views of audit assurance: recent evidence on the expectation gap. *Journal of Accountancy*, January: 60-66.

Lohr, S., 1997. Be Paranoid. Hackers are out to get you. *New York Times Ondisk*.

Hackenbrack, K. 1992. Implications of seemingly irrelevant evidence in audit judgement. *Journal of Accounting Research* (Spring): 54-76.

Marczewski, D. C. and Akers, M. D. 2005. CPA's Perceptions of the Impact of SAS 99. *CPA Journal*, vol 75, pages 38-40.

McQuaid, J. M, et. al., 2000. Tools for Distributed Facilitation. *HICSS 2000*.

Nigrini, M. 1999. I've Got Your Number. *The Journal of Accountancy* 187: 5.

Palmrose, Z-V, Richardson V.J., and Scholz, S., 2004. Determinants of Market Reactions to Restatement Announcements, *Journal of Accounting and Economics*, vol 37, pp. 59-89.

Pincus, K. V., 1989. The Efficacy of a Red Flags Questionnaire for Assessing the Possibility of Fraud. *Accounting, Organizations, and Society*. 14: 1/2, pages 153-163.

Reed, M. R., and Pence, D. K. 2005. Detecting Fraud in Financial Statements: The use of digital analysis on Analytical review procedure. *Journal of Forensic Accounting*. Volume 6, pages 135-146.

Smith, S. G. 2005. Computer Forensics: Helping to Achieve the Auditor's Fraud Mission? *Journal of Forensic Accounting*, vol 6, pages 119-134.

Winters, A. J. and Sullivan, J. 1994. Auditing for fraud: Perception vs. reality. *1994 Kansas Audit Symposium*.

Zimbelman, M.F. 1997. The effects of SAS No. 82 on auditors' attention to fraud risk factors and audit planning decisions. *Journal of Accounting Research (Supplement*): 75-97.

Waller, W. S., and Zimbelman, M.F. 2000. An empirical assessment of external validity in audit research on the dilution effect. *Working Paper*. Brigham Young University.

The opinions of the authors are not necessarily those of Louisiana State University, the E.J. Ourso College of business, the LSU Accounting Department, or the Editor-In-Chief.