Ensuring the Consistency of Self-Reported Data: A case study

(Practice-Oriented)

Hassan Davulcu
davulcu@xsb.com

Jennifer Jones
j.jones@xsb.com

Robert Pokorny
pokorny@xsb.com

Chris Rued
c.rued@xsb.com

Terrance Swift
tswift@cs.sunysb.edu

Tatyana Vidrevich
tatyana@xsb.com

David S. Warren
warren@cs.sunysb.edu

XSB, Inc., Stony Brook, NY

Abstract: Much data available from on-line databases is self-reported. Because of this the quality and consistency of web available data is often suspect. This paper presents the XSB, Inc. supply chain diversification tool, a service directory that collects and presents data from the Small Business Administration website and a number of web based subscription registries of minority vendors. The diversifier uses a number of automated tools including extraction, classification, standardization, and matching to present information on minority suppliers and classify them to the NAICS, or the *North American Industry Classification System*. The system enhances data quality by automatically collating data from multiple sources with semantics based on the NAICS taxonomy and using methods of statistical quality control to quantify the quality of the data presented.

INTRODUCTION

Much data of interest to organizations is self-reported: this includes data from surveys of customers, from registration of members or users, and so on. In collecting self-reported data, say over the world-wide web, the kinds of edit checks available are limited by the fact that data entry must be easy – otherwise members or customers may neglect to report the data of interest. As a result, the designer of "forms" for self-reporting often faces a choice between ensuring the quality and consistency of data, and ensuring that a sufficient amount of data is collected. In this paper we present a case study of how a suite of data cleaning techniques, including web agents, data standardization, data matching, and textual classification can be used to help the quality of semi-structured self-reported data. The structure of the paper is as follows. We motivate the problem by a tour through the screens of the XSB Diversifier, a directory for finding minority or women-owned suppliers using amalgamated data from the world-wide web. Details of the techniques

that ensure various aspects of the XSB Diversifier data quality are presented in Section 2. Finally, Section 3 presents metrics concerning the precision of these techniques over the XSB Diversifier's data.

1. THE XSB DIVERSIFIER: AN INTEGRATED SERVICE DIRECTORY FOR MINORITY VENDORS

The XSB Diversifier helps large organizations increase their supplier bases by finding vendors that are owned by women or minorities. Registries of small suppliers can be found in various places around the web. Those used by the XSB Diversifier included the Small Business Administration or SBA (http://pro-net.sba.gov) and four other publicly available and subscription registries. When an organization registers at one of these sites, standard information is entered such as the company name and address, phone, contact person, etc. In addition, a short profile is entered about the types of products or services delivered by the organization. Finally, to enable searches, some of the registries ask the organization to provide a set of *NAICS* classification codes describing its industrial classification. NAICS, or the *North American Industry Classification System*, (http://www.census.gov/epcd/www/naics.html) is a taxonomy of industry classifications for use in Federal reports, designed by the U.S. Census Bureau.

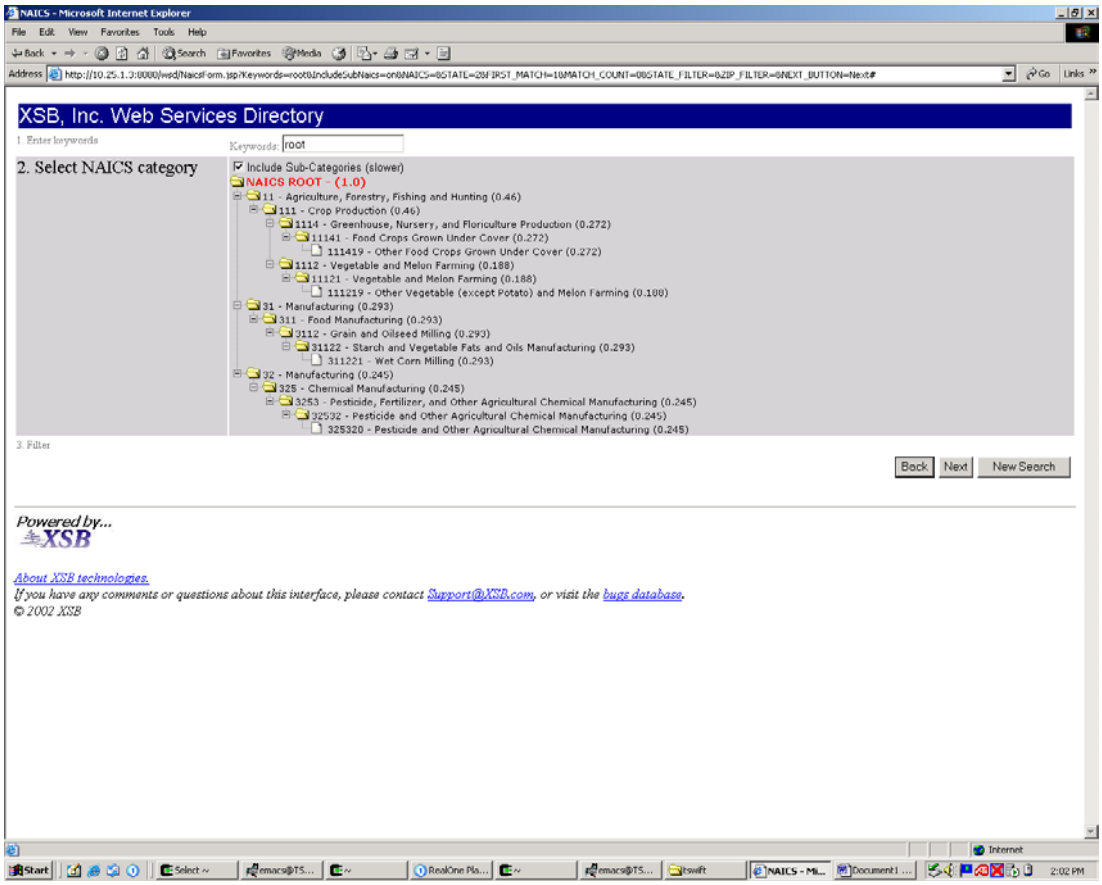


Figure 1: Entering NAICS Classifications

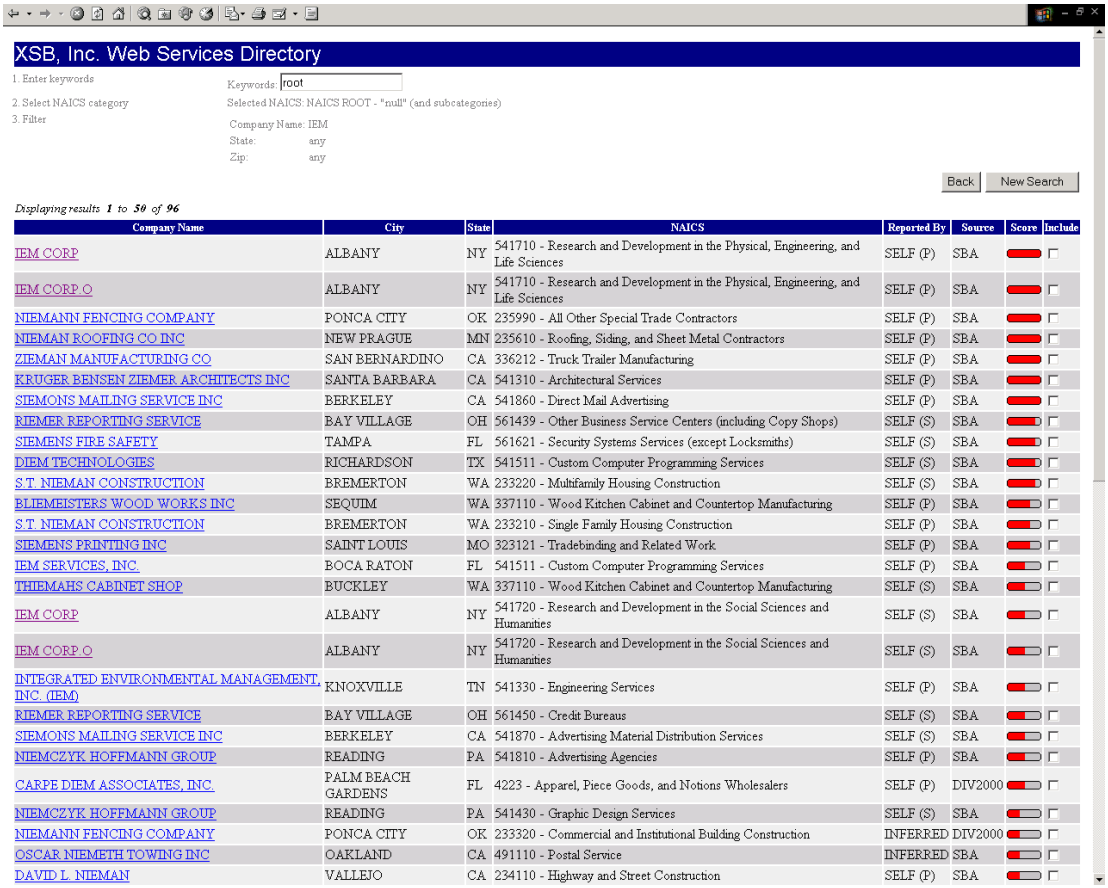
While the above sites have a wealth of useful data, their data is often poorly classified. For instance, querying one of the sites for organizations under the NAICS category "EGG PRODUCTION, CHICKEN" returns a furniture store. The main purpose of the XSB Diversifier is to extract data from the above sites; amalgamate it and resolve

inconsistencies from the various sources; and to organize it in a way so that it is easily searchable by NAICS code, organization name, address, and other attributes.

One of the first input screens from The XSB Diversifier shows the top of the NAICS taxonomy. This screen allows a user to restrict a minority supplier search to various economic classes. The search is, of course, hierarchical in that selecting a general category searches not only the category itself, but all descendant categories in the taxonomy. In Figure 1, the root of the NAICS taxonomy is selected, denoting an unrestricted search.

A subsequent screen allows further refinement of a search based on NAICS keywords, as well as on an organization's name, and various address components. In The XSB Diversifier, searches are made to a SQL Server database that is periodically refreshed by XSB Inc's XRover agents (Section 2.1), which harvest information from the various registries, feeding this data to other data cleaning routines and creating as an end product a *coherent view* of the various data elements. As a next step in our example we enter the keyword 'IEM' for the organization name, leading to the screen in Figure 2.

Figure 2 shows various companies harvested from the registries that match the search criteria. For this example, we click on the first entry 'IEM CORP' to view detailed information, leading to the screen in Figure 3.



XSB, Inc. Web Services Directory

1. Enter keywords
2. Select NAICS category
3. Filter

Keywords: root

Selected NAICS: NAICS ROOT - "null" (and subcategories)

Company Name: IEM
State: any
Zip: any

Back | New Search

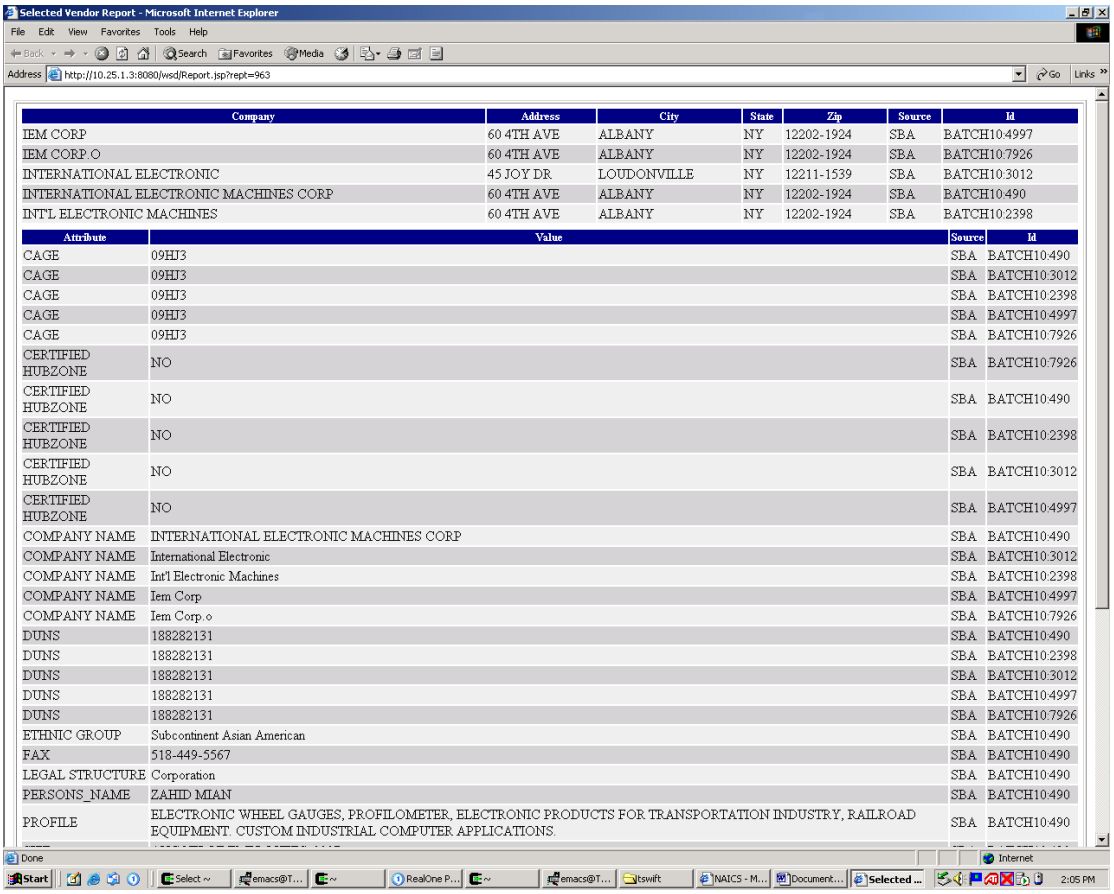
Displaying results ***1*** *to* ***50*** *of* ***96***

Company Name	City	State	NAICS	Reported By	Source	Score	Include
IEM CORP	ALBANY	NY	541710 - Research and Development in the Physical, Engineering, and Life Sciences	SELF (P)	SBA		
IEM CORP O	ALBANY	NY	541710 - Research and Development in the Physical, Engineering, and Life Sciences	SELF (P)	SBA		
NIEMANN FENCING COMPANY	PONCA CITY	OK	235990 - All Other Special Trade Contractors	SELF (P)	SBA		
NIEMAN ROOFING CO INC	NEW PRAGUE	MN	235610 - Roofing, Siding, and Sheet Metal Contractors	SELF (P)	SBA		
ZIEMAN MANUFACTURING CO	SAN BERNARDINO	CA	336212 - Truck Trailer Manufacturing	SELF (P)	SBA		
KRUGER BENSEN ZIEMER ARCHITECTS INC	SANTA BARBARA	CA	541310 - Architectural Services	SELF (P)	SBA		
SIEMONS MAILING SERVICE INC	BERKELEY	CA	541860 - Direct Mail Advertising	SELF (P)	SBA		
RIEMER REPORTING SERVICE	BAY VILLAGE	OH	561439 - Other Business Service Centers (including Copy Shops)	SELF (S)	SBA		
SIEMENS FIRE SAFETY	TAMPA	FL	561621 - Security Systems Services (except Locksmiths)	SELF (S)	SBA		
DIEM TECHNOLOGIES	RICHARDSON	TX	541511 - Custom Computer Programming Services	SELF (S)	SBA		
S.T. NIEMAN CONSTRUCTION	BREMERTON	WA	233220 - Multifamily Housing Construction	SELF (S)	SBA		
BLIEMEISTERS WOOD WORKS INC	SEQUIM	WA	337110 - Wood Kitchen Cabinet and Countertop Manufacturing	SELF (P)	SBA		
S.T. NIEMAN CONSTRUCTION	BREMERTON	WA	233210 - Single Family Housing Construction	SELF (P)	SBA		
SIEMENS PRINTING INC	SAINT LOUIS	MO	323121 - Tradebinding and Related Work	SELF (P)	SBA		
IEM SERVICES, INC.	BOCA RATON	FL	541511 - Custom Computer Programming Services	SELF (P)	SBA		
THIEMAHS CABINET SHOP	BUCKLEY	WA	337110 - Wood Kitchen Cabinet and Countertop Manufacturing	SELF (S)	SBA		
IEM CORP	ALBANY	NY	541720 - Research and Development in the Social Sciences and Humanities	SELF (S)	SBA		
IEM CORP O	ALBANY	NY	541720 - Research and Development in the Social Sciences and Humanities	SELF (S)	SBA		
INTEGRATED ENVIRONMENTAL MANAGEMENT, INC. (IEM)	KNOXVILLE	TN	541330 - Engineering Services	SELF (P)	SBA		
RIEMER REPORTING SERVICE	BAY VILLAGE	OH	561450 - Credit Bureaus	SELF (S)	SBA		
SIEMONS MAILING SERVICE INC	BERKELEY	CA	541870 - Advertising Material Distribution Services	SELF (S)	SBA		
NIEMCZYK HOFFMANN GROUP	READING	PA	541810 - Advertising Agencies	SELF (P)	SBA		
CARPE DIEM ASSOCIATES, INC.	PALM BEACH GARDENS	FL	4223 - Apparel, Piece Goods, and Notions Wholesalers	SELF (P)	DIV2000		
NIEMCZYK HOFFMANN GROUP	READING	PA	541430 - Graphic Design Services	SELF (S)	SBA		
NIEMANN FENCING COMPANY	PONCA CITY	OK	233320 - Commercial and Institutional Building Construction	INFERRED	DIV2000		
OSCAR NIEMETH TOWING INC	OAKLAND	CA	491110 - Postal Service	INFERRED	SBA		
DAVID L. NIEMAN	VALLEJO	CA	234110 - Highway and Street Construction	SELF (P)	SBA		

Figure 2: Summary Organization Information for Match

Note that in addition to the company selected, 4 other companies also show up as aliases of IEM CORP. To derive these aliases, raw name and address data is *standardized* to a canonical format, and given companies are matched to determine whether they are new to the system or are variant entries of a known company (these techniques are described in Section 2.2).

In Figure 3 all 5 variant companies happen to come from the SBA registry, but standardization and matching is even more critical in resolving conflicts for entries from different registries [1] Accumulated data from SBA is also reported for each of the variant organization entries. Of particular interest here is the organization's self-reported profile seen at the bottom of Figure 3, indicating that IEM CORP is a type of electronics manufacturer.



Company	Address	City	State	Zip	Source	Id
IEM CORP	60 4TH AVE	ALBANY	NY	12202-1924	SBA	BATCH10:4997
IEM CORP.O	60 4TH AVE	ALBANY	NY	12202-1924	SBA	BATCH10:7926
INTERNATIONAL ELECTRONIC	45 JOY DR	LOUDONVILLE	NY	12211-1539	SBA	BATCH10:3012
INTERNATIONAL ELECTRONIC MACHINES CORP	60 4TH AVE	ALBANY	NY	12202-1924	SBA	BATCH10:490
INT'L ELECTRONIC MACHINES	60 4TH AVE	ALBANY	NY	12202-1924	SBA	BATCH10:2398

Attribute	Value	Source	Id
CAGE	09HJ3	SBA	BATCH10:490
CAGE	09HJ3	SBA	BATCH10:3012
CAGE	09HJ3	SBA	BATCH10:2398
CAGE	09HJ3	SBA	BATCH10:4997
CAGE	09HJ3	SBA	BATCH10:7926
CERTIFIED HUBZONE	NO	SBA	BATCH10:7926
CERTIFIED HUBZONE	NO	SBA	BATCH10:490
CERTIFIED HUBZONE	NO	SBA	BATCH10:2398
CERTIFIED HUBZONE	NO	SBA	BATCH10:3012
CERTIFIED HUBZONE	NO	SBA	BATCH10:4997
COMPANY NAME	INTERNATIONAL ELECTRONIC MACHINES CORP	SBA	BATCH10:490
COMPANY NAME	International Electronic	SBA	BATCH10:3012
COMPANY NAME	Int'l Electronic Machines	SBA	BATCH10:2398
COMPANY NAME	Iem Corp	SBA	BATCH10:4997
COMPANY NAME	Iem Corp.o	SBA	BATCH10:7926
DUNS	188282131	SBA	BATCH10:490
DUNS	188282131	SBA	BATCH10:2398
DUNS	188282131	SBA	BATCH10:3012
DUNS	188282131	SBA	BATCH10:4997
DUNS	188282131	SBA	BATCH10:7926
ETHNIC GROUP	Subcontinent Asian American	SBA	BATCH10:490
FAX	518-449-5567	SBA	BATCH10:490
LEGAL STRUCTURE	Corporation	SBA	BATCH10:490
PERSONS_NAME	ZAHID MIAN	SBA	BATCH10:490
PROFILE	ELECTRONIC WHEEL GAUGES, PROFILOMETER, ELECTRONIC PRODUCTS FOR TRANSPORTATION INDUSTRY, RAILROAD EQUIPMENT. CUSTOM INDUSTRIAL COMPUTER APPLICATIONS.	SBA	BATCH10:490

Figure 3: Detailed Information for IEM CORP, Pt. 1

Figure 4 shows the rest of the screen for IEM CORP, including its NAICS codes. Most of these NAICS codes are self-reported and concern a broad research classification, but the bottom code, OTHER ELECTRONIC COMPONENT MANUFACTURING, has been inferred from its profile via a textual classification system (Section 2.3). The assignment of electronic component manufacturing classification to IEM CORP means that IEM CORP will be an element of searches for electronic manufacturers, whereas otherwise it would be missed in these searches. Once the user has the information from the previous screens he or she can contact the company for further information.

[1] For copyright reasons, this example only shows publicly available data from the Small Business Administration.

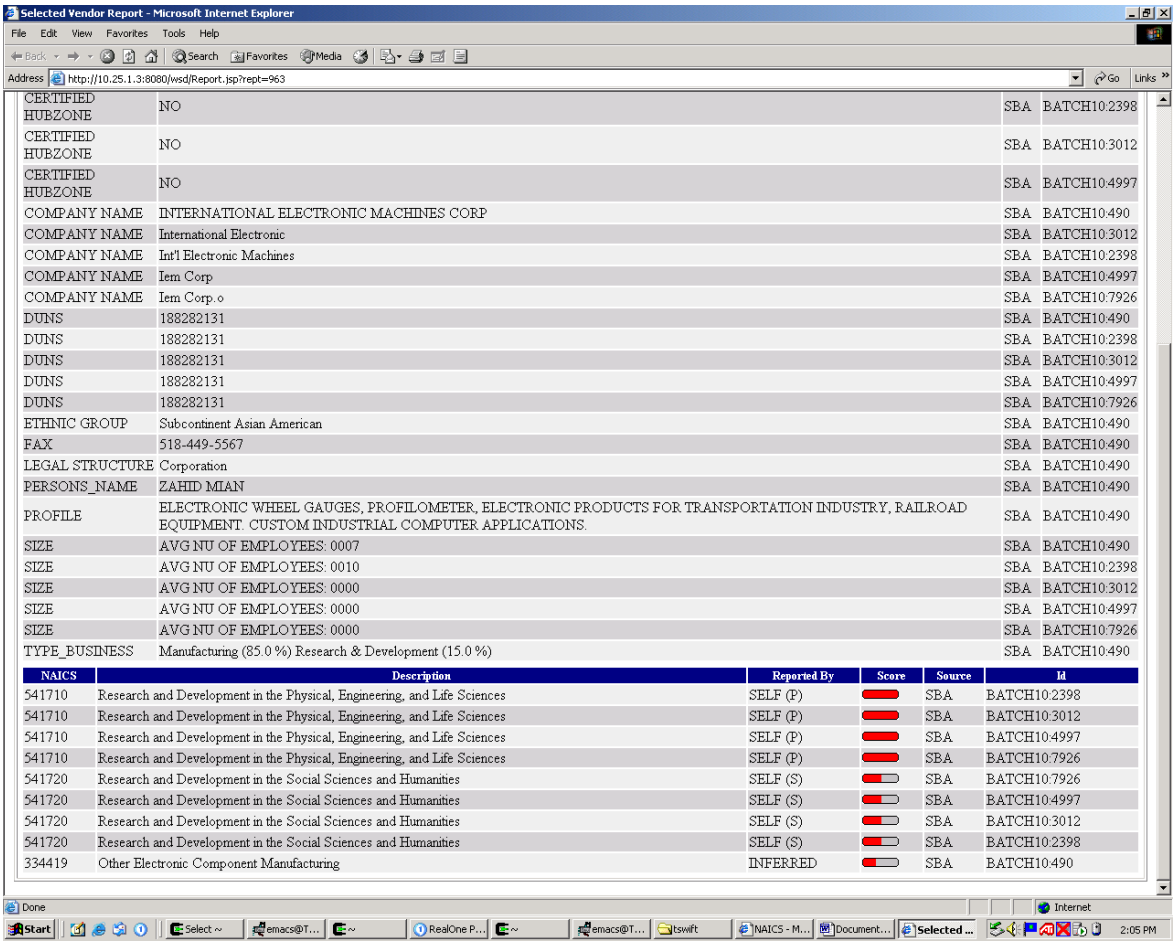


Figure 5: Detailed Information for IEM Corp, Pt. 2

Supporting the relatively simple screens shown are several novel techniques for ensuring data quality to which we now turn. The data about the various minority vendors lies in semi-structured web pages of the various registries, often behind several layers of forms. The XRover agents are responsible for accessing this data and putting it into the SQL server databases. In registries where NAICS codes are available they are self-reported by the registering company. In other registries self-reported NAICS codes may be missing or incomplete. An Ontology-Driven Classifier is used to determine codes from textual descriptions supplied by the registering company. Thus, NAICS codes can be supplied when they are missing and compared to self-reported codes to broaden the areas of coverage of an organization. Finally, the data harvested from the web is of uneven quality, and arises from heterogenous sources. Data standardization and matching are used to improve the quality of name and address data.

Before turning to brief discussions of these technologies, we present some summary statistics about The XSB Diversifier. In all, The XSB Diversifier contains harvested information concerning 180,903 company records, including duplicate records. Of these, about 83% had self-reported NAICS classifications, mostly from the SBA site. For the rest, their primary classification was inferred from the Ontology-Driven Classifier..

2. TECHNOLOGIES UNDERLYING THE XSB DIVERSIFIER

2.1 XRover Web Agents

A first step in implementing The XSB Diversifier is obtaining data from the various registries of minority vendors, which is done with XRover web agents. Each XRover agent is defined by a SitePlan, which is a data structure consisting of PageMaps and Actions that interconnect the PageMaps. Each PageMap consists of a set of regular expressions indicating which parts of a web page contain meaningful information to be extracted, along with basic formatting information for output. Actions can be defined to follow links or to fill out forms. When an XRover agent is launched, the agent interprets the SitePlan performing the actions indicated, formatting extracted data and inserting it into SQL server tables as it goes.

Figure 5 shows part of the front screen for the SBA registry, where users may query using a variety of mechanisms. Upon making a query, the site returns tabular information about the organizations that satisfy the criterion specified in the query. Companies in the SBA registry were gathered by general agent queries on a state-by-state basis. Thus, conceptually the SitePlan for this site consisted of two PageMaps. The first PageMap consisted of regular expressions indicating the positions of various elements of the form, and the second consisted of regular expressions indicating how to extract information from the tables returned for the queries. They were connected by an Action indicating that the State and other information was to be entered in a form. Additional SitePlans allow XRover agents to navigate the other registries mentioned in Section 1.

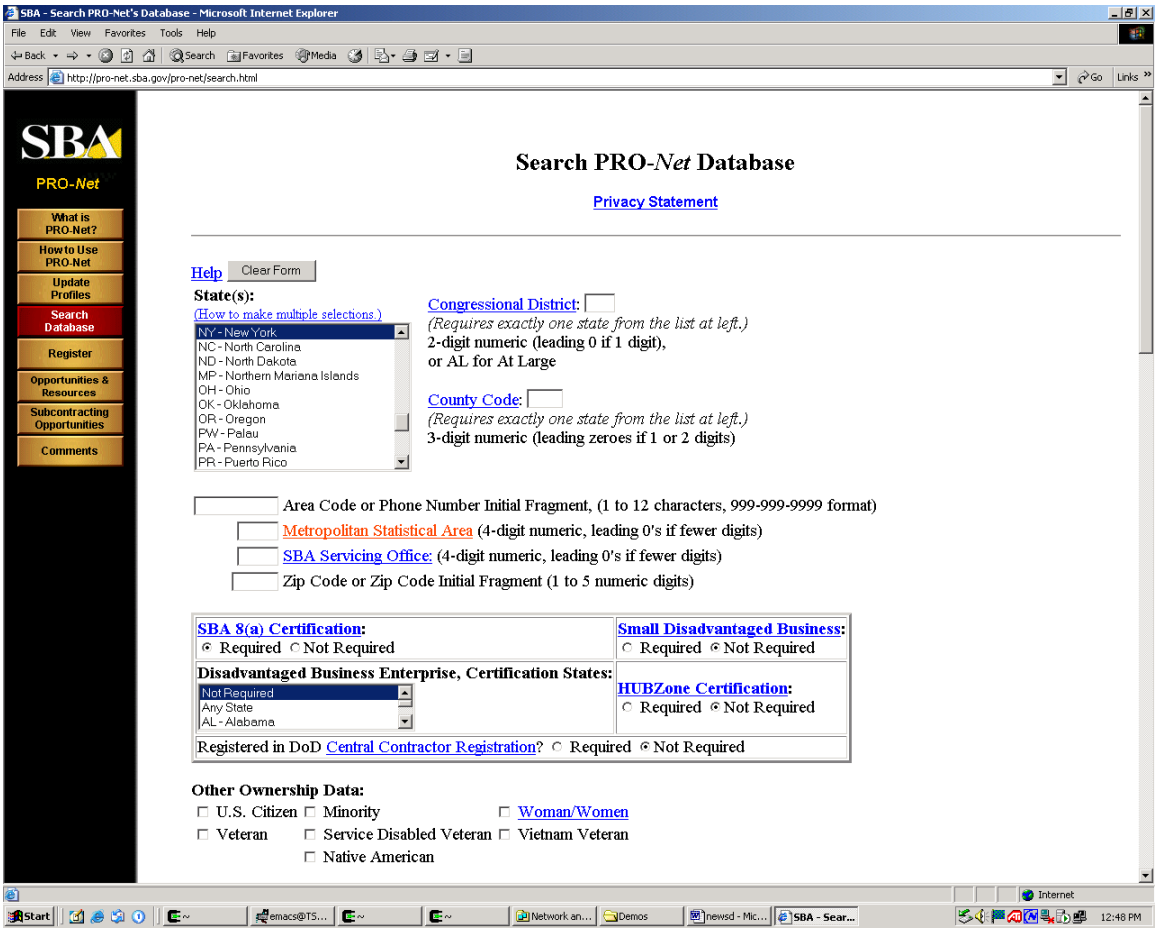


Figure 6: Input screen for a PublicVendor Registry

While the navigation of this site is fairly simple, involving only two pages, an advantage of XRover technology is its use of regular expressions for navigating sites. Many changes can be made to the input page, e.g. by adding or rearranging information in it, without invalidating the SitePlans used by the agents.

Data extracted by the agents is used by the standardization and matching routines, and by the Ontology-Directed Classifier, all of which are described below.

2.2 Standardization and Matching

Once data has been extracted into fields it may contain numerous errors and inconsistencies. For example, data for 'IEM CORP' extracted by the XRover agents contains the name 'IEM CORP' and 'IEM CORP O' both of which should be treated in the same way. Furthermore, although the agents extract various address fields, these fields may contain street and post office addresses in various positions, and other information may be enjambed between fields. In order to recognize when the various data sources refer to the same entity standardization and matching techniques are used.

The techniques used for data standardization have been presented elsewhere in the literature [2], and we present here only a brief description. Data standardization takes unstructured or semi-structured data as input and outputs fully structured data. In The XSB Diversifier, data standardization is performed using XSB Prolog [4] whenever the SQL Server database is refreshed by the XRover agents. Data standardization consists of three main phases. First, an input data field (or fields, when data elements may span fields) is broken up into tokens in the usual manner. Second, the tokens are deterministically turned into super-tokens, which form semantically meaningful units. For instance, the tokens 'NEW' and 'YORK' may be transformed into the meaningful token 'NEW YORK', while 'ONE FOOT LONG' may be transformed into the Prolog term 'measure(1,foot,length)'. Once super-tokens are generated, they are available for semantic tagging indicating, for instance, that 'NEW YORK' is a state or city, and that 'measure(1,foot,length)' is a dimension.

In name and address standardization, the super-tokens are then parsed via *preference logic grammars* (described in [2]). Intuitively, the need for preference logic grammars arises from the ambiguity of name and address data. For various text strings, it is often ambiguous when an organization name ends and an address begins; or how to separate different address elements from one another (e.g. a string such as 'IEM BROADWAY NEW YORK' might arise in data sources that do not clearly enforce distinction between a company name, street address and city). Preference logic grammars are used to resolve ambiguities in the following manner. Rather than attempting to resolve ambiguities within the grammar productions themselves, ambiguous productions are written, and preference clauses indicate which of (an ambiguous) set of parses to retain. Typically, the use of preference logic grammars can significantly reduce the amount of standardization code, making such code easier to maintain.

Once names and addresses have been standardized, the standardized data can be used to identify duplicate entries for the same organization. Duplicates arise both because companies may register at multiple registries and also register at a particular registry more than once. Duplicate records are identified by matching components of the name, address and phone number. A score based on a weighted sum of these components matches indicates how closely two entities in the database match each other. Statistical sampling techniques for quality control as described in [1] are then used to determine an Acceptable Quality Level (AQL) for each score number. The AQL is a quantitative measure of how many defective matches to expect per hundred matches made. So if a match receives a score of 6 and a score of 6 has an AQL of 10, one would expect that match to be accurate 90% of the time.

This matching algorithm for identifying is an enhanced version of the one described in [3]. It has been improved by adding a component for matching on telephone number. This is used on both the reported contact phone number and fax number when available.
The matching algorithm has also been enhanced by adding negative scoring in the various components when there is an obvious mismatch. Previously, a zero was assigned both when two component values being matched were different and when one or both of the component values were null. Now a –1 is assigned when the component values are different.

2.3 Ontology-Driven Classification

The XSB Diversifier uses Ontology-Driven Classification in two ways. As mentioned above, these inferred classifications augment self-reported classification data for minority vendors. As used here, Ontology-Driven Classification addresses what we may call the object classification problem. As input, we are given a short (say under 256 characters) textual description of an object, *O*, along with a taxonomy of nodes, each of which also have short textual descriptions. Elements of the taxonomy are considered to be sets of objects, and the ordering of the taxonomy is considered as set inclusion. The classification output is a set of taxonomy nodes considered to be the "best" guesses of the sets of which *O* is a member. For instance, in the The XSB Diversifier example shown in Section 1, the object corresponds to the IEM CORP, and its description is found in the Profile field of Figure 2.

> "ELECTRONIC WHEEL GAUGES, PROFILOMETER, ELECTRONIC PRODUCTS FOR TRANSPORTATION INDUSTRY, RAILROAD EQUIPMENT, CUSTOM INDUSTRIAL COMPUTER APLICATIONS"

NAICS constitutes the taxonomy, and examples of the textual descriptions of its nodes are provided in Figure 1, where the description matched the NAICS node "OTHER ELECTRONIC COMPONENT MANUFACTURING".

In general, Ontology-Driven Classification has several stages:

1. *Taxonomy Token Weighting*. Taxonomy descriptions are tokenized and super-tokens are created in much the same manner as described in Section 2.2 (this step may include elimination of certain tokens that are irrelevant for classification, such as the tokens "the", "a", etc. These super-tokens can be used to correct obvious misspellings, to account for common abbreviations, and so on. For each super-token *T*, the non-normalized weight of *T* is taken to be total occurrences of all super-tokens in the taxonomy divided by the total occurrences of T in the taxonomy. This weighting gives higher weight to tokens that occur less frequently in the taxonomy, which are likely to be more useful for classification.

2. *Node Weighting for Descriptions*. The object description is supertokenized, and each a weight is derived for each node in the taxonomy as a function of the supertokens in the description that match the nodes description, their position in the descriptions, contiguity of tokens, and so on.

3. *Weight Propagation and Normalization* Given the semantics of taxonomies as being ordered by set-inclusion, the classification weight of a node *N* is taken to be the weight of *N* as determined in Step 2, together with the sum of the Step 2. weights of all of its children. In this step previous weights are propagated and then normalized so that the weight of the root of the taxonomy equals 1.

4. *Determining the "Best" Node.* Based on the results of step 3, an essentially greedy search starts at the root and descends the tree to determine the "best" match for the object description based on the node's normalized match weight of Step 3. Users of Ontology-Directed classification may use various cutoffs to determine when the descent should stop along with various relaxations of the greediness of the descent.

It can be shown that the normalized weights produced by Step 3 form a probability measure when the taxonomy graph is a tree. For the example from Section 1, the match was made to "OTHER *ELECTRONIC* COMPONENTS" because of the match to the italicized token as well as matches to the nodes ancestors, "SEMICONDUCTOR AND OTHER *ELECTRONIC* COMPONENT MANUFACTURING", and "*COMPUTER* AND *ELECTRONIC PRODUCT* MANUFACTURING". The match thus represents a compromise between specificity – to find a node as far down on the taxonomy as possible – and correctness.

The Ontology-Driven Classifier can be tuned in two basic ways. First, the classification algorithm is heavily dependant on the weights of super-tokens as determined in step 1. By tuning the super-tokenizer which is applied both to taxonomy nodes and to textual descriptions, the various weights of nodes can be affected. Second, training items can be provided. These are descriptions that are pre-classified to their correct taxonomy node. Training items

are treated as if they extended the taxonomy, being taxonomy nodes that are immediate children of the node to which they classify. Then the processing proceeds on this "larger" taxonomy. The only difference in treatment is that in step 4, when the best node is determined, these new are excluded from being chosen. So training descriptions "pull" similar descriptions toward themselves. The XSB Diversifier used SIC code descriptions as training items. SIC is the precursor to the NAICS taxonomy, and there exist generally available mappings from SIC codes to NAICS codes. We assumed that a SIC description was an object classified to the corresponding NAICS taxonomy node, and so used them as training items. This resulted in approximately 20,000 training items being used for the approximately 2500 taxonomy nodes.

Ontology-Driven Classification is based on textual descriptions, and is domain-independent apart from rules used for super-tokenization. It differs from other classifiers in its use of the hierarchical structure of the taxonomy. For example, a word that appears in 100 distinct taxonomy nodes all clustered in one general portion of the taxonomy, will pull items to that portion of the taxonomy. Words that appear uniformly across the entire taxonomy will not have much affect the classification process, essentially introducing noise. So the structure of the taxonomy itself influences the effect that particular words have; words related to the semantic concepts of the taxonomy will have greater affect than unrelated words.

3. DATA QUALITY METRICS

There are many issues of data quality when amalgamating data from heterogeneous sources. The main issues that has been investigated for The XSB Diversifier are the matching process for identifying duplicate records, and the accuracy of the Ontology-Driven Classification.

The matching process can be measured using the AQL as a reasonable quantitative metric. The results of assigning AQLs to each match score level are shown in figure 6. It can be seen from this figure that there is a sharp break between good matches and bad matches. All scores of 5 and below are bad matches and all scores of 6 and above are good matches. This allows automatic identifying of duplicates with about 98% accuracy.

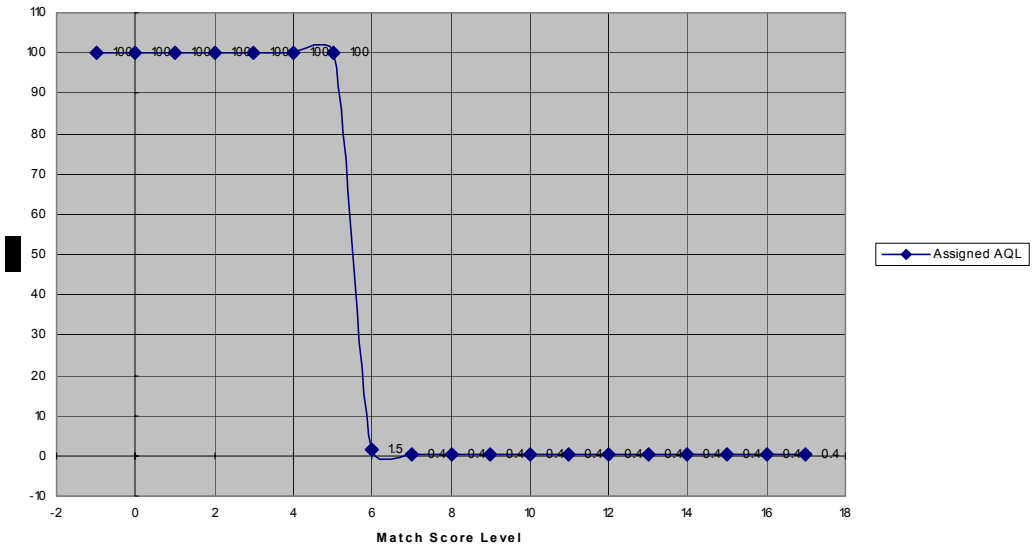


Figure 6: Assigning AQLs to Match Score Levels

The quality of the results of the classifier depends heavily on a number of properties of the taxonomy, the quality and typicality of training items, and the data. For classification to a taxonomy, an inferred classification is deemed correct if it is the same classification a user would make: an inferred classification is incorrect even if it is more or less specific than the classification made by an expert. In various applications inferred classifications have been correct between 60% to 95% of the time. Tuning with appropriate replacements can improve performance to some degree. The greatest benefit comes from good training descriptions. The classifier can be incrementally tuned by running it on a small batch of test data, reviewing the results, hand-correcting the errors, using those corrected classifications to add the items as training, and iterating.

The techniques used to assign AQLs to the matching process can in principle be used to assign an AQL to the classification process. The methodology would be to select a statistically significant sample of classified items and classify these manually comparing the manual result to the result of the classifier. Although this has not yet been done with the NAICS classification in The XSB Diversifier, it has been tried on other classification problems and has given guidance in selecting training items to improve the classification. This will be applied to the XSB Diversifier Classifier in the near future.

4. Conclusions

The XSB Diversifier illustrates several points about the use of data quality techniques in commercial systems. First, as is well known, techniques for insuring data quality are needed for information drawn from heterogenous sources and even non-trusted second-party sources. Second, the process of insuring data quality need not be manually done by an analyst: the extraction, standardization, matching and classification used in the XSB Diversifier are automatic processes and each of these techniques can be configured for a given application via a graphical tool or a documented API. This allows the creation of integrated presentations such as the XSB Diversifier that rely on data from multiple heterogeneous web sources.

There is manual labor involved in building the agents to extract data from the web, tuning a classifier to the semantics of a particular domain described in a taxonomy such as NAICS, and validating the quality of the data transformation processes such as matching multiple records and classifying text descriptions. Once this effort has been completed, the integrated presentation of data can be automatically maintained as the underlying data changes.

Some of the algorithms that underlie the XSB Diversifier's data quality techniques are sophisticated and their development and integration benefited from recent academic research. Incorporating them in applications like the XSB Diversifier illustrate their commercial potential for enhancing data quality from myriad web sources.

REFERENCES

[1] ANSI/ASQC Z1.4-1993. Sampling Procedures and Tables for Inspection by Attributes. American Society for Quality Control, Milwaukee, Ws. 1993.

[2] Cui, B. and T. Swift. Preference Logic Grammars: Fixed-Point Semantics and Application to Data Standardization. *Artificial Intelligence*. To appear..

[3] Pokorny, L.R. "Assigning a Quality Measurement to Matching Records from Heterogenous Legacy Databases: A Practical Experience", in Proceedings of the 2000 conference on Data Quality, Cambridge, Ma. Pp 70-75

[4] The XSB Programmer's Manual, v. 2.6. http://xsb.sourceforge.net. 2002.