Comparison of Deduplication Methods between Two Immunization Information Systems

By

Mark C. Goodman

A Capstone Project

Presented to the Department of Medical Informatics & Clinical Epidemiology and the Oregon Health & Science University School of Medicine

In partial fulfillment of the requirements for the degree of

Masters of Biomedical Informatics

June, 2010

School of Medicine

Oregon Health & Science University

Master of Biomedical Informatics

CERTIFICATE OF APPROVAL

This is to certify that the Capstone Project of

Mark C. Goodman

"Comparison of Deduplication Methods between Two Immunization Information Systems"

Has been approved

_____

Judith R. Logan, MD, MS

_____

Date

Table of Contents

Tables

Examples

Acknowledgements

I would like to thank my advisor, Dr. Judith Logan, for her assistance, guidance and inspiration during my graduate studies. I would also like to thank Andrea Ilg and Diane Doctor for their support navigating the requirements for the curriculum.

Additionally, I would like to extend my thanks to Mary Beth Kurilo for the opportunity to intern at The Department of Human Services (DHS), as well as Donald Dumont, Amber Crandell, and Andrew Osborn for their assistance gathering the information needed to complete this project.

Abstract

Objective: To compare the deduplication rate and accuracy of demographic record deduplication processes implemented by Oregon Immunization Alert's current and new deduplication systems.

Methods: Evaluate the capabilities of the two demographic deduplication systems using a test set created by the Centers for Disease Control (CDC) with known duplicate and non-duplicate records. Measure the duplicate record detection rate and accuracy along with the amount of time required by human intervention. Compare this to the CDC's recommended deduplication rate of 90+% and accuracy rate of 98+%.

Results: In the evaluation of the current system, we were able to deduplicate 84% of the duplicates in the test set. The accuracy of the current system was 99.7%. This process took a total time of 3.5 hours. In the evaluation of the new system, we were able to deduplicate 93% of the duplicates in the test set. The accuracy of the new system was 97.25%. The deduplication process took a total time of 12 seconds.

Conclusion: Data quality is extremely important for the Oregon Immunization Alert Registry. The current system, which requires a great deal of human intervention, is a legacy system which is an amalgamation of several processes, performed by different staff members, all working with in-house tools that have become less and less effective as the workload has increased. The new system is an automated system which requires far less human intervention to maintain data quality and handle the deduplication process. In comparing the test results for both systems, the superior performance improvement of the new system should meet the registry's current and future data quality requirements while significantly lowering the need for human intervention.

Introduction

Immunization registries have been promoted as a critical tool that can encourage increased vaccination coverage. The benefits of an immunization registry (also known as an Immunization Information System, or IIS) are believed to be so substantial that the Centers for Disease Control and Prevention established a national objective to have at least 95% of US children, age 6 years or younger and that have two or more vaccinations, to be included in a fully operational population-based immunization registry.[1]

Since its inception in 1998, the Oregon Immunization Alert registry has grown from a small organization collecting data from a handful of healthcare providers to one with over 1900 public and private providers. The registry now contains over 50 million vaccination records on over 5 million children and adults.

Encouraging provider participation is crucial to creating an effective registry. For the Oregon Immunization Alert registry, encouraging the public providers (e.g. county health departments and government entitlement programs such as Vaccines for Children) to participate was not as difficult as encouraging the private providers. There are several barriers to private provider participation: the time and cost needed to collect and send in the data; the belief that this data collection should be publically funded; and primarily, concerns that overall, the data might not be comprehensive and of high quality.[2-14]

In many states there is a legislated mandate that providers must submit their patient immunization records to the state's immunization registry. In Oregon, however, this is not the case. In order to encourage providers to participate and overcome their reluctance to do so, the Alert registry geared their policies and procedures toward making it as easy for providers as possible. The first step was to allow them to submit

their immunization information in many different formats. This has ranged from paper forms and bar code labels all the way to electronic formats such as spreadsheets and HL7 electronic messages. All of these formats require a great deal of customization and human intervention by the registry personnel before inclusion in the registry.

An obstacle to capturing a complete immunization history is the fact that immunizations are a ubiquitous healthcare procedure. The public can receive immunizations in numerous places: from their primary care doctor, a pharmacy, a county clinic, at the hospital, at their school, their place of employment and numerous other facilities. In addition, health insurance providers keep records of when an immunization is administered. Any of these may be submitted to the Alert registry. So not only does the Alert registry have records from numerous providers, they also have records of immunizations from non-providers, and may receive multiple reports on the same immunization event. This leads to another quality assurance problem: duplicate records. Duplicate records happen both for the demographic record and for the immunization record.

Oregon Alert's procedures for dealing with data quality and data deduplication have gradually improved over the past 12 years. Because of constrained resources, it has been necessary to take a piecemeal approach, adding fixes and process changes to accommodate the growing demand. The Alert team handles most data quality and duplication cleanup issues post-extraction, i.e. when the data is requested for various purposes, such as for CDC reports, clinic requests, etc. Although multiple processes and procedures have had to be created to clean the data post-extraction, once that data has been scrubbed and deduplicated, it is viewed as accurate and comprehensive by the various groups that request it.

Managing data quality and data duplication has become an enormous task for the Alert registry and they have launched a large scale project to address this task by moving

from their current, heavily burdened system, to a newer system, Alert IIS, with services provided by Hewlett Packard (HP, formerly EDS). HP has a great deal of experience working with immunization registries, having already converted fourteen other state registries over to their systems.[3] They also handle data quality and cleansing as well as data deduplication as part of their service. These processes are very different from the current data cleaning processes used by the Alert registry. The model used by HP is designed to perform data validation and data deduplication as data comes into the system, rather than after the data is extracted from the system. HP also has stronger constraints on the data to help reduce the number of anomalies within the stored records.

Because of the size and importance of this project, as well as the fact that the two systems are very different in design, the Alert team wants to ensure that the transition is a smooth one. Not only do they want the impact on their providers to be minimal, they also want to make sure that the new data deduplication processes and procedures do not reduce the richness of the data.

This paper will compare the deduplication algorithms and procedures used by both systems, Oregon Immunization Alert and Oregon Alert IIS, and compare each system's effectiveness at deduplicating demographic information. It will provide an overview of the issues related to data quality, an explanation of the concepts of data quality and deduplication, and finally, a comparison of the deduplication processes implemented by both systems.

Background

What are immunization registries?

Immunization Registries (a.k.a. Immunization Information Systems or IISs) are confidential, computerized information systems that collect and combine vaccination data from multiple healthcare providers. They also may produce reminder/recall vaccination notifications and vaccination coverage assessments for children within a specific geographic area.[4] Many IISs have additional capabilities such as vaccine management, adverse event reporting, birth-to-death vaccination histories, and interoperability with other health information systems.[5] In addition, a fully operational IIS can help prevent duplicate vaccinations, limit missed appointments, reduce vaccine waste, and reduce staff time required to produce or locate vaccination records or certificates.[6]

An example of IIS effectiveness is the streamlining of administrative tasks required for both health departments and private practices that are associated with large entitlement programs, such as Vaccines for Children (VFC). The IIS benefits include documentation of doses administered, tracking of under-immunized children, and consolidation of vaccination records from multiple providers (public and private), all functions needed for VFC program funding.[7] Newer uses of IISs include helping in response to emergencies (including pandemic influenza, bioterrorism, or disasters)[8], monitoring the impact of vaccine shortages, monitoring uptake of new vaccines (e.g. H1N1), increased tracking for adult immunizations and linking to other health information systems.[6]

There have been significant improvements in IIS use since the CDC launched its national objective for 95% child participation in a registry. More than one half of the nation's children are now tracked in a population-based IIS. However, significant issues still need

to be resolved, such as ensuring sustainable funding for IISs, maintaining data quality while increasing the patient participation rate, increasing private provider participation, and expanding IISs so that they can communicate with each other and with other health information systems. This includes communication with electronic medical record (EMR) systems and health information exchanges (HIE).

Provider Participation

A challenge for many state IISs is encouraging private healthcare providers to participate in the registry. Many states have legislation that compels providers to participate in an IIS; however, Oregon is not one of them. Provider participation is crucial for IIS success. If a registry cannot encourage a provider to participate, then a complete vaccination history will be more difficult to assemble without gaps in the patient's record.[9] For the provider, IIS use offers some advantages and can assist with managing inventory, maintaining complete vaccination records, and prompting practices to remind patients when vaccinations are due.[10]

In most cases, once parental consent is given, a child is entered in an IIS at birth, often through linkage with an electronic birth certificate or on the first visit with a health care organization. Demographic and immunization data are recorded at enrollment and electronically transferred from the provider office to the central registry database. At each immunization encounter, the vaccination event is recorded by one or more organizations (e.g. the health care provider and the insurance provider may both record the event).[11]

An IIS, in many instances, can provide a more thorough immunization history than any single clinic.[9] Several factors may account for this finding, such as the fact that over 20% of children in the United States move during the first 2 years of life and an even higher percentage change health care providers.[12] Therefore a registry that captures immunization information from multiple providers can have a more complete history than an individual immunization provider, particularly in highly mobile populations.[13]

Provider Attitudes about IISs

Since provider participation is so important to the success of any immunization registry, studies have been performed in an attempt to determine the benefits and costs associated with participation. In a study done by Christakis, et al, on providers' perceptions of an immunization registry, 77% percent of RN/NPs, 60% of pediatricians and 47% of family physicians ($p < 0.001$) responded that they thought immunization registries represented the "best chance to solve the documentation problem." Only 15% of respondents reported that they preferred parental record keeping to registries.[14]

In the same study from 1999, 57% percent of RN/NPs, 61% of pediatricians and 43% of family physicians reported that the incompleteness of registry data presented a barrier to their using one. Forty-one percent of RN/NPs, 30% of pediatricians, and 18% of family physicians reported that lack of training was a barrier. Twenty-two percent of RN/NPs, 42% of pediatricians, and 29% of family physicians reported that costs were a barrier ($p = 0.01$). Time was a barrier for 41% of respondents.[14]

Table 1, from Biomed Central in 2006 illustrates factors that providers consider when determining whether to participate in an IIS. In both this and the previously mentioned analysis, the most influential factor for provider participation is to have an accurate, complete, and consolidated vaccination record.[15] This is a challenge for any

Table 1: Influence of Factors on Decision to Participate in Registry

	Very Influential	**Somewhat Influential**	**Not Influential**
Need to consolidate records for patients who receive vaccines at multiple sites	63%	28%	9%
State mandates participation*	54%	26%	20%
Nursing/administrative staff in favor of participation	40%	35%	25%
Ability to use registry to monitor immunization rates	33%	36%	31%
Medicaid/health plan mandates participation	32%	31%	37%
Compatibility of registry technology with office computers	27%	37%	36%
Availability of technical assistance/training from state	25%	41%	34%
Expected cost/staff time required for data input	21%	45%	34%
*This option given only in states with an existing legislative mandate			

Source: Clark SJ, Cowan AE, Bartlett DL. Private provider participation in statewide immunization registries. BMC Public Health 2006 Feb 15;6:33.

organization participating in health information exchange that involves consolidating health records from various sources.

Data Quality: Constraints, Validation and Cleansing

Data quality for a database begins with defining, validating and maintaining data integrity. The database schema and integrity constraints govern permissible values on individual fields as well as their relationship to each other. Since cleansing data after it has entered the system is an expensive process, preventing the input of invalid or "dirty" data in the first place is an essential part of data quality.[16]

There are three basic strategies that can be used to prevent dirty data in database records. First, be sure to design an appropriate schema and data dictionary for the database. [17] Second, create integrity constraints for each field. Finally, enforce those integrity constraints by incorporating validation checks within the applications used for data entry.[18]

Database Design Constraints

In their paper, "Data Cleaning: Problems and Current Approaches", Rahm & Hong define a difference between dirty data related to schema violations and dirty data related to instance violations.[19] In Tables 2 and 3 are examples of data that violates either the schema level or the instance level of a record. The current Alert Immunization Registry contains data with many of these anomalies in one form or another.

Table 2: Schema Design Level Problems

Object	Problem Type	Dirty Data Example	Reason/Remark	Resolution
Field	Illegal values	DOB=13/30/1970	Values outside the range for month	Avoid free text fields
Field Attribute	Uniqueness violation	Record1(SSN='123456789') Record2(SSN='123456789')	Uniqueness for SSN violated	Set the UNIQUE constraint
Field Trigger	Violated attribute dependencies	Age=22, DOB=12/02/70	Age should = (current date – birth date)	Set trigger on age field
Field	Referential Integrity Violation	Record3(Dept_Num=122)	However, dept 122 is not defined	Add a "CHECK" constraint

Adapted from: Rahm E, Hong HD. Data Cleaning: Problems and Current Approaches. IEEE Techn. Bulletin on Data Engineering, Dec. 2000. Available at: http://wwwiti.cs.uni-magdeburg.de/iti_db/lehre/dw/paper/data_cleaning.pdf. Accessed 03/30, 2010.

Table 3: Instance Input Level Problems

Object	Problem Type	Dirty Data	Reason/Remark	Resolution
Field	Magic Numbers	phone=999-999-9999	Random input when actual data is unknown (dummy values or null)	Create a specific "unknown" type
	Misspellings	city="Protlund"	usually typos, phonetic errors	Verify value prior to input
	Cryptic values, Abbreviations	experience="B"; occupation="DBProg."	User error	Use predefined code sets
	Embedded values	Name="J. Smith 12/02/70 New York"	Multiple values entered in a single attribute (free form field problem)	Use normalization To keep fields as atomic as possible
	Misfielded values	City="Oregon"	User error	Verify prior to input
Record	Violated attribute dependencies	City="Redmond", zip=77777	City and zip code should correspond	Verify prior to input
Record Type	Word Transpositions	Name1="J. Smith" Name2="Smith, J."	Usually in a free-form field	Use normalization to keep fields as atomic as possible
	Duplicated records	Name1="John Smith" Name2="J. Smith"	Same patient represented twice due to data entry error	Use deduplication strategies prior to input
	Contradicting records	Name1="John Smith","DOB=12/02/70" Name2="Jack Smith","DOB=12/02/70"	Sometimes Jack is used for John, this is ambiguous if it is the same person	Use deduplication strategies prior to input
Source	Wrong reference	Name1="John Smith","vacc=DTT"	There is no vaccine coded as DTT	Use predefined code sets

Adapted from: Rahm E, Hong HD. Data Cleaning: Problems and Current Approaches IEEE Techn. Bulletin on Data Engineering, Dec. 2000. Available at: http://wwwiti.cs.uni-magdeburg.de/iti_db/lehre/dw/paper/data_cleaning.pdf. Accessed 03/30, 2010.

Table 4: Data Validation Techniques[20]

Data Validation Techniques	Examples
Avoid Default Values	- Using 1/1/1900 as a default value. - Unknown, UNK, "baby", "baby boy" etc. are all codes for a newborn with an unknown first name. - Using 999-99-9999 or any other number as a default SSN
Have an explicit "Unknown" option for the user	If the user does not have that information, then have a preset option of "unknown".
Minimize the number of free text fields	Ex. If the field records an SSN, do not allow any out non real numbers e.g. 123-45-6789, do not allow text in a number field etc.
Avoid using one field for multiple purposes	Ex. First name field that says, "MC Jones" which means "male child of Jones" etc.
Build in error checks	- Is the date an actual date value? - Does the vaccination event take place before the date of birth?
Collaborate with the electronic data providers	- Find field level mappings. - Avoid mangling data in transmission. - Validate data before loading it.
Provide real-time record matching	Before a record is added to the system, the system checks whether it is already entered.
Try to link records by a reliable id wherever possible	Reduces the impact of duplicate records

Data Validation

The best way to avoid problems related to integrity constraints is to initiate data entry validation checks prior to the input of the record into the database. This is accomplished by implementing data validation rules. Data validation is "the process of ensuring that a program operates on clean, correct and useful data".[21] Table 4, illustrates common input checks that, if properly implemented, are another step to ensuring good data quality.

Data Cleansing

The final step in ensuring data quality is data cleansing. Data cleansing (also called data cleaning or data scrubbing), is the process of locating and correcting errors and inconsistencies in order to increase the data's accuracy. In a single database there can be numerous problems with data quality such as misspelled words, missing information, and outdated and/or obsolete data.[19] When multiple data sources are integrated, such as in a data warehouse, the need for data cleansing increases significantly. In order to maintain consistent and accurate data, both the consolidation of distinct data representations and the removal of duplicate information are necessary.

The process of data cleansing can be broken down into four basic steps. The first step is to audit the data. This is where data is reviewed in an effort to locate anomalies in each field. Second, create a workflow specification, where any anomalies are examined and a sequence of operations is defined to fix them. The third step is workflow execution, when the repairs designed in the previous step are executed on the data sets. As a final step, post-processing and controlling is when the results of the execution are verified and any errors that could not be corrected automatically are corrected manually, at which point new integrity constraints are put in place to prevent dirty data from entering the system with any future updates.[22] Essentially, this is the process that is underway as the Oregon Immunization Alert registry moves from its current system to the new Alert IIS system hosted by HP. See table 5 for examples of popular cleansing methods.

Once the data cleansing process has been accomplished, the next step for a heterogeneous database (i.e. a database that accepts data from various sources) such as an immunization registry, is to resolve issues around duplicate records. This applies to both duplicate records that currently exist in the database as well as preventing the entry of duplicate records in the future.

Data Deduplication

The records contained in an immunization registry must be sufficiently complete and accurate to be used effectively when guiding patient care.[23] The existence of duplicate

Table 5: Popular Cleansing methods[16]

Type	Explanation
Parsing	Performed for the detection of syntax errors. Checks whether a string of data is acceptable within the allowed data specification.
Data Transformation	Data is transformed into a format expected by a given application. Often done as part of ETL. Also, done as part of normalization of table values.
Statistical Methods	Analyzing the data to look for values that are unexpected and erroneous. Then using statistical methods, setting the values to an average or some other statistical value
Duplicate Elimination	Deduplication of redundant records

patient records in an immunization registry is an ongoing problem. As a registry grows, it receives data from numerous sources; this can lead to an increasing number of inaccurate, missing or duplicate data. Most registries have found it necessary to allocate substantial resources to assure data quality.[24]

There are two main types of duplicate records that occur within immunization information systems. The first type is a duplicate immunization record, for example, a vaccination record that is entered twice, perhaps days apart, once when the shot was administered and again when it is billed. The second type is a duplicate demographic record which is where records of a patient's demographic information differ in one or more fields, such as name, address, or telephone number.[25] The focus of this section will be on demographic record deduplication.

Duplicate records are a potential problem in any demographic database, but particularly in an immunization information system that has multiple contributing sources of patient information. There are many factors that add to the complexity of the deduplication process. For example, there are numerous common first names, a name may have multiple spellings (e.g. Gayle or Gael), a legal name may change, addresses frequently change, many places no longer record the social security number for privacy reasons, many children do not have social security numbers, optional data fields are left blank, or the original data may contain errors (e.g. typographical errors).[23]

Several software applications as well as manual review processes have been designed to help with deduplication. These use probabilistic and deterministic matching algorithms on demographic data fields such as date of birth, first and last name, address, guardian's name, etc. in an attempt to identify possible duplicate records.[26] Regardless of how well any given immunization registry's processes perform, deduplication is essentially an attempt to reach an idealized goal. It is impossible to determine with absolute certainty when all records are deduplicated.[27]

Deduplication encompasses several sub-processes. Duplicate and/or fragmented data must be prevented from entering the system. Redundant information needs to be removed from the database on an ongoing basis. Assurances must be made that queries and updates apply to the correct data.[28] These are non-trivial processes because redundant information is often difficult to discern, correct data may exist in numerous records, and the same data can be represented in various forms. The task of reducing duplicate demographic information in integrated patient health information is generally divided into three sub- categories, data standardization, record matching and record combining.

Data Standardization and Transformation

To effectively match demographic records, the individual data fields must be separated into atomic datum in order to facilitate more efficient and accurate comparisons. This is particularly important for free text fields that hold information such as addresses, where there are several bits of information separated by spaces (e.g. street name and number, city, state, zip). Another task of standardization is to ensure that the format of the data is consistent between the records, so they can be compared accurately (e.g., dates can be encoded in several different ways mm/dd/yyyy or yyyymmdd etc.).[29]

Names. The name fields are widely used in record matching; however, standardizing names is not a trivial task due to the many possible inconsistencies that can occur. For example, names can be misspelled during data entry, some people use aliases, initials and/or nicknames, and names are entered in various forms, e.g. with hyphenation from cultural/ethnic differences.[30]

The basic approach to dealing with name comparisons is straight-forward string matching and edit-distance comparisons. String matching consists of comparing one string to another and determining if there is an exact match. This is a very fast operation to perform. Edit-distance comparisons focus on string similarity. String similarity can be specified in terms of distance, where distance is defined as the number of edits it would

require to turn one string into another string. If two strings are *more* similar, the distance between the edits of the strings will be less. For example, the distance between "Joe" and "Joe" is zero, because no edits are required to convert a string into itself. The edit distance between “Joe” and “Joey” is one, because a deletion of one letter “y” is required to make the strings identical.[31] Edit distance comparisons allow for variations in name spellings, as well as typographical errors and work well for weighted match algorithms. String matching is fast and works well for exact match algorithms.

There are numerous phonetic algorithms used to encode names by the sound of their enunciation, with comparisons made based on this encoding. The most well known of these is Soundex. The goal is to encode homophones (i.e. names that sound the same but are spelled different, such as Kym and Kim) to achieve the same representation, thus being matched despite minor differences in spelling. The algorithm mainly encodes consonants; a vowel will not be encoded unless it is the first letter.[32] Soundex was designed for English names and is understandably less effective with non-English names. Another obstacle with the Soundex approach is dealing with multiple or hyphenated first names. For example, in one record the first name may be “Maria Jessica”, in another record it may be “Maria” for the first name and Jessica for the middle name. Hyphenated last names present a similar problem.

Addresses. This field should be parsed into its individual components. Usually, the city, state, country and zip are already separated. However, the street name, number and apartment number should be standardized. For example: “Eight First Avenue, #5h” can be converted to the standardized “8 1st Ave Apt. 5H”. Putting addresses into a standard format and then verifying the address's validity can be a difficult process.

Other Demographics. Other demographics such as mother's name and/or maiden name, telephone number, Social Security Number, and Medicaid Number may also be used to help match records. These can be valuable fields when evaluating whether or

not two records are duplicates. However, many of these fields are optional and contain no data, so they are usually used to confirm probable matches that have been established by name, birth date and address comparisons.[33]

Record Matching

Matching is the process of finding records that already exist in a registry that might correspond to a new record being entered. Matching can be divided into two sub-problems, finding candidate records and clustering records into groups of matching or potentially matching records.[28] Matching records can be a very resource intensive task depending on the application. Table 6 lists project types and examples where matching is used.

When Matching is Performed. Record matching typically happens at two points, at the front-end, i.e. during data input when data is entering the system, and/or at the back-end, i.e. after data input, by processing records in batches looking for duplicates that already exist in the database. Front-end matching can be either interactive or non-interactive. As an example of interactive front-end matching, when a user attempts to

Table 6: Types of Matching Applications[26]

Algorithm Types	Examples
One-Time Matching Projects	- Single instance removal of duplicates from a single database. No deduplication on an ongoing basis. - Single instance deduplication when combining two or more databases. No deduplication on an ongoing basis. - Checking the database for duplicates during manual entry.
Ad-hoc Matching of Many Different pairs of databases	- Frequent requirement of research institutions. Merging numerous Excel spreadsheets for a variety of studies.
Maintaining the Integrity of a Single Database	- First deduplicate the database and then provide an input interface to make sure that no new duplicate records are added to the database. - *Example*: Before any new record is added to the database the client system queries the database to determine if the record information already exists. If the system returns a match then the new record is linked to the preexisting record.
Linking Multiple Databases Using a Master Patent Index	- Similar to single database problem. Before a patient is added to any single database, check in the master index to find its master index id. If there is no id, then generate one.

enter a person's information, they could first be required to do a search on the name and address. The user would then be presented with possible pre-existing matches for the person, allowing them to link the new information to a pre-existing record. For an automated non-interactive input system, the record is compared during input, and if a potential match is found, the record can be automatically deduplicated or put aside for manual review.

The back-end approach is most often a batch job that is run periodically. Each record in the database is examined and compared to see if it matches any others. Usually this approach also involves organizing records into groups or clusters, each being a set of potential matches. The advantage of a back-end solution is that the system can automatically find and resolve large numbers of duplicates in batches. This allows large integrated systems to process numerous records without human intervention. Many matching systems incorporate both front-end and back-end duplicate detection. An integrated health system such as an immunization registry can benefit from a combination of both paradigms.

Types of Matching Algorithms. There are four basic types of matching algorithms -- single-field comparison, multi-field matching, rule-based matching, and probabilistic record matching with machine learning.

Single-field Comparison. Algorithms based on single field comparison attempt to find potential matching records based on a single field in the table. This method is very quick but also very limited in terms of how well it performs at finding meaningful matches. With identifying fields that are unique, however, it can be very effective.[34] For example, Social Security Number and Medicaid Number have a standard format and are unique to each person, so they can be an effective means for determining a duplicate record. However, they are present in only a few childhood immunization records. Yet, when used in combination with other matching techniques, single-field matching is a very efficient first step in identifying duplicate records.

Table 7: Types of multiple field comparisons[26]

Comparison Function	Explanation
Relational	Includes =, <,> and != comparisons.
Partial String	String comparison functions that limit the comparison to a specific number of characters, e.g. just the first five letters of the last name.
Containment	Functions that can determine whether a field value is either fully or partially contained within another field.
Ranges	Functions that determine whether a numerical or data field value is within some specified range of another field. E.g. the vaccination dates are within +/- days of one another
Edit-distance	Functions that determine the minimum of number of editing operations (e.g. insert a character, delete a charter, or replace a character) necessary for making two values the same. Edit distance is a good approximation of keystroke errors.
Phonetic Matching	Soundex comparisons match strings (typically names) with different spellings but similar sequences of characters. They do this by removing all vowels & vowel sounding consonants (except for the first letter). Then encode the rest of the remaining characters as a series of digits. There are numerous other phonetic encoders besides Soundex and many database systems provide direct support for information retrieval based on Soundex comparisons.

Multi-field Comparison. There are numerous multi-field matching algorithms. However, they all attempt to find matches by comparing two or more fields, for example, by using a combination of last name, date of birth and gender. Often the system is configurable with regards to both which fields are used and which comparison function is performed on each field (e.g. digits vs. alpha characters).[35] Table 7 lists various types of multi-field comparison strategies.

Related to the use of multi-field comparisons in deduplicating demographic records is the fact that every record may have varying amounts of incomplete demographic data. So not only do algorithms need to deal with how to compare fields, they also need to pick fields that have a very high likelihood of being present in an immunization record. When deduplicating records, there is a trade off between the goals of maintaining a high true positive rate while at the same time keeping the false positive rate as low as possible. A true positive match is correctly matching two or more records that describe the same person. A false positive match is when two or more records, that do not describe the same person, are incorrectly matched and reduced to a single record. A false positive is very difficult to find once two records have been deduplicated. This can

lead to one or both people being under-immunized, which is taken very seriously by the immunization registry.

A 2001 study by Miller, Frawley and Sayward[33] was designed to determine the most advantageous fields on which to match. In this study they found that the most powerful strategy was one they called "ORing high-value ANDed" pairs. High value fields, such as last name, first name, and date of birth were present in 99+% of their records. Gender and address where present in 96% of the records. They combined the high value pairs together, for example, last name AND date of birth OR last name and first name. They found that when they performed matching on different combinations of the pairs, i.e. ORing them together, they could achieve a very high true positive rate of 96% or more, while keeping the false positive rate at or below .12%. See table 8 for high value pair combinations.

Another form of a multi-field comparison algorithm accounts for the frequency of a field value. The more frequently the value occurs in the data set, the less weight that particular match will have in the overall summation. For example, if the name "Michael" appears in more records than "Dweezil", it is less discriminating and carries less weight. Use of frequency can improve the accuracy of a multi-field matching algorithm, but it also requires more computational overhead.

Table 8: Advantageous Multi-Field Matching Combinations

Data Set Combinations	Avg. True Positives % (per dup)	True Positive%	Avg. False Positives % (per dup)
LN & (DOB | FN | SSN)	1.58	93.4	0.09
FN & (DOB | LN SSN)	1.48	87.7	0.04
DOB & (LN | FN | SSN)	1.57	93.0	0.08
DOB & (LN | FN | SSN | Addr | MinN | MMN)	1.59	93.9	0.12
LN & DOB | LN & FN | LN & SSN | DOB & FN | DOB & SSN	1.65	97.6	0.11

Adapted from: Miller PL, Frawley SJ, Sayward FG. Exploring the utility of demographic data and vaccination history data in the deduplication of immunization registry patient records. J.Biomed.Inform. 2001 Feb;34(1):37-50

Rule-based Matching. Similar to multi-field matching algorithms, rule-based algorithms can involve an assortment of comparison functions and multiple field comparisons. However, unlike multi-field matching algorithms they do not determine a match score by combining the individual field comparisons. Rather, they apply a set of decision rules of the form `IF <comparison> Then <action>.` For example, a rule for matching on SSN first, then, if that is not possible, matching on a combination of Soundex comparison and date of birth might be written as follows:

```
IF record1.SSN = record2.SSN THEN match
Else IF Soundex_Compare (record1.Lname, record2.Lname) AND
      record1.DOB = record2.DOB THEN match
```

The advantage of rule-based algorithms over multi-field matching is that they can short-circuit the comparison steps by testing the most discriminating rules first, e.g. comparing SSNs. This can dramatically speed up the overall matching time, especially for large databases with millions of records.[36] Single-field and multi-field algorithms can be combined with rule-based matching. One approach might be to use the single-field method for fields that have a high rate of accuracy, such as birth date or SSN, then, if there is a match, use multi-field matching for other fields in the rest of the record.

Machine Learning (Probabilistic Record Matching). For the algorithms mentioned so far -- single/multi-field and rule-based approaches -- three tasks need to be done to allow the algorithms to be used effectively: determine which fields are the most useful, formulate the best way to compare those fields, and decide how much impact those fields have when determining a match. Another approach to solving these problems is using machine learning algorithms. A common machine learning algorithm is probabilistic record matching. Probabilistic record matching (Naïve Bayes algorithm) is a form of multi-field matching which relies on weights for the various fields.[37] It uses a training process, where pairs of records are labeled as being matching or non-matching as determined by human review. Then, weights are assigned to each field based on statistics calculated from the labeled records. The records are processed by the system,

with their true match/no-match status known. The system then determines a match using the previously configured rules and compares the result to the true answer. If it is correct it then reinforces the rules in the system, if not, it adjusts the rules slightly to compensate for the mistake.

In order to be effective, a great deal of training data needs to be processed through the system. Once enough data has been processed by the system, it should be able to accurately determine matching pairs for data that are similar to the training set. Usually a system based on probabilistic record matching also has a low-to-high threshold range. A score at or above the threshold can be considered a match and the records deduplicated. Any score below the range can be considered a non-duplicate. Any score that is within the range can be labeled as a possible match and set aside for further investigation by human reviewers.[38]

Blocking and Scoring. Most matching schemes combine single/multiple field and rule-based matching, plus probabilistic matching algorithms into a combination known as "blocking and scoring". The first step is the blocking phase, in which records are compared looking for an exact match on a particular field or combination of fields, such as birth date and/or last name. Once a cluster of potential matches has been identified, the scoring algorithm (using probability matching) assigns a numerical value to the likelihood that the potential duplicate records represent the same entity.[39]

An advantage of the probabilistic-matching approach is the use of a threshold range where the system can be tuned to allow a desired balance between recall and precision. Recall is related to the true positive rate, and pertains to how well the system does at finding the duplicate records from all of the records. Precision is related to the false positive rate, as it pertains to how accurate the system is when it determines that a set of records are duplicates. Immunization registries are willing to accept some loss of recall (which translates to more potential matches being missed or needing human

review) as a trade-off to increase the precision (ensuring that any automatic matches are truly duplicates).

Other Matching Considerations. As one can see, record matching is more complex than it first appears. Tables 9-11 list other issues to consider when designing a matching process.

Table 9: Other Matching Considerations[27]

Overall Quality of the Data Source	Example
Is the data from a high quality source?	- Government Vital Records e.g. birth certificate, SSN card
Is the data from a low quality source?	- Forms filled out on the internet (w/o verification). - Unverified Information (e.g., from parents memory).
Is the data up to date?	- Some fields change less: SSN, DOB - Some fields change more: Home address
Are fields blank or filled with invalid data?	- Sometimes the name field can be filled in with "Baby" etc. or "unknown"

Table 10: Challenges for Matching[20]

Unique Matching Obstacles	Example
Unusual data fields	- Matching on guardian such as foster parent or grandparents. - Matching on physician or clinic.
Using unusual criteria for a match	- Identify children who may reside at multiple addresses during the year (e.g. divorced parents). - Identify twins, have a field to indicate birth order, twins can be especially difficult to discern.
Unique challenges specific to a particular source	- In some cases, a clinic may report vaccinations as given at the end of the month, rather than the actual immunization date.

Table 11: Matching Accuracy[27]

Acceptable Level of Accuracy	Example
False negatives: missing duplicate records	- System indicates that two records differ when in fact they are the same. - Ok for immunization records, accept the risk of over-immunization.
False positives: matching non duplicate records	- System indicates that two records are a match when in fact they are not. - This is considered not acceptable for immunization records as it may lead to under-immunization, which can have severe public health consequences.
Work level needed to achieve the required accuracy	- There is a trade off between the level of accuracy of automatic matching and the amount of records marked for human review. - Some cases do not require a high level of accuracy, such as data being used for statistical purposes; minimal errors do not make a significant difference.

The Final Deduplication Step - Record Merging or Linking

After a pair or set (cluster) of matching records has been found, the final step is to perform either consolidation of those records or removal of the duplicate data. For systems that employ a real-time data input and lookup method (i.e. a user interface for interactive front-end matching), the new record can be linked directly to the current information. Any record updates can be made in real-time to the actual record, negating the need for deduplication. For back-end systems, potential duplicates with a high confidence level can be automatically deduplicated. Those that fall within a "possible match" range can be marked for manual review.[27]

In practice, deduplication can be accomplished in several ways. One option is by merging the data of the records in a duplicate cluster into a single record. Another is linking together all of the records available, without deleting any records, to form a complete, if redundant, history (a cluster). This way all of the submitted information is available upon retrieval of the primary record. A third option of deleting all but one of the records is not practical for medical data due to the risk of losing vital information.

Record Merging. Record merging entails taking the individual data items from each of the records in a duplicate cluster and merging that data into a single record. This approach can be cleaner than record linking in terms of database management, but it is not simple to achieve. Some of the obstacles to merging data are standardizing the datum values between different records and resolving potential conflicts between the data items. Resolving conflicts is especially cumbersome when trying to decide which piece of data is more accurate. Additionally, there may be loss of historical data when records are merged.[40] Because of the potential loss of data, record merging may not be the best choice for demographic data, where keeping records of change in address, for example, may be important. However, the method may be a better fit to immunization data since an immunization event is static, and deduplication involves information from different sources about the same event.

Record Linking. Record linking is a good choice for merging duplicate records once the potential duplicates have been found using any of the matching techniques. Record linking involves setting up a key between the potential duplicate records. Sometimes this is done by adding another field to a table or creating another table to establish the virtual link between records, usually known as a Master Patient Index (MPI).[41] The MPI links together all of the records of a single person.

Overview of the Current Alert Deduplication Process

The deduplication method used by the Oregon Immunization Alert registry is a combination of semi-automated and manual-review processes. These processes are used for new data coming into the system as well as outgoing data that is requested by clinics or used for research purposes. See Appendix A for an illustration of the current Alert output processes.

The current Alert deduplication process performs linked-based merging. In this case, there is no actual deletion or deduplication of data. Instead, this system uses an MPI that associates the records between the tables. Using this approach, a thorough and accurate representation of a child's immunization history is captured. See appendix B for a diagram of the database schema.

All of the deduplication processes are performed on an extracted copy of the data stored in temporary tables. Duplicate matches are then resubmitted with their updated MPI number (IDAlias.ID). Every demographic record in the database has a unique identifier (Demographic.IDS) and it is associated with one entry in the master patient index table (IDAlias.ID). Keep in mind that each person in the registry can have one or more demographic records associated with them. The individual demographic records are identified by an IDS field. This field is created for every record that comes into the registry. It is a combination of a unique organization id that is assigned to every provider and the medical record number given to a patient within the organizations medical record system. For example: the Alert staff assigns the organization id 'KP', to Kaiser Permanente. Kaiser has their own medical record system and assigns a medical record number to an individual patient,'123456'. In the Alert database this record would have the combined IDS as KP123456. Each IDS is associated with a single ID in the IDAlias table (the master patient index). Each IDS can be associated with only one ID and an ID

can be associated with many IDS (i.e. IDS=demographic and immunization records, ID=MPI for a single person).

The IDAlias table tracks these associations between a single person and numerous demographic and immunization records. See Appendix C for a diagram of the master patient index table linkages.

Source processing & Initial Inspection

Incoming data is processed asynchronously, in batches, daily or weekly. There is a lag time between when data comes in and when it is uploaded into the database. This is due to the fact that organizations are submitting data in many different formats. All of these formats need to be pre-processed into a single format that can then be imported into the database.

Import/Export System

After the data validation and transformation process has been performed, the data is processed through what is called the "Import/Export" filter. Many organizations submit the same set of records every time. The incoming demographic records are compared to records in the database using the IDS field. By checking the IDS field, matching records can be compared quickly. If the incoming record has new information the record in the database is updated, if not, the record is discarded. All records without a pre-existing IDS move on to the next step. See Appendix D for an overview of the Alert input/output data flow.

Once the processing of records with matching IDS fields is complete, the remaining records are designated as non-duplicate and potential duplicate records. Non-duplicates are separated from potential duplicates by comparing the all of the incoming records to records in the database. It uses a rule-based algorithm that employs single-field blocking on an exact match of date of birth and a multi-field match string comparison on first and last names. The rule is as follows:

```
IF new_record.dob=current_record.dob AND
String_Compare (new_record.first_name, current_record.first_name) AND
String_Compare (new_record.last_name, current_record.last_name) THEN
Place Record in the Demographics Resolve Table
```

Any records that do not match exactly on all three criteria are determined to be new records and are entered into the database. Any records that match exactly on birth date and a close string match on first and last name are also entered in the database and a copy of the record is placed in the demographics resolve table. See Appendix E for a diagram of the import/export system.

Manual Review Processes

Resolve Process. At this point copies of the new potential duplicates are in the demographics resolve table. The staff at the registry uses an application called "Resolver" to review the potential duplicate records. Resolver creates links between the potential duplicate demographic records in the resolve table and demographic records in the live database. It then presents the potential matches together in a single user interface. Then the manual review steps are repeated for all potential duplicates in the resolve table. If the potential matches are accepted as duplicates, Resolver creates a new master patient index and assigns the ID value to each matching record in the Alert database. See Appendix F for a diagram of the resolve process.

Auto-Merge Process. The Auto-Merge software is used to deduplicate records already in the database. A set of records are extracted from the database, stored in a temporary table and then Auto-Merge performs the following steps:

1. Removes punctuation, spaces, hyphens etc. from the first and last names.
2. Convert the first and last names of the potential matches to Soundex encoded variables.
3. Compare the extracted table entries to potential matches in the database, based on exact match of birth date, and converted first and last names from step 2.

4. Rank them with a potential match score from 1-6, (1 = least likely, 6 = most likely a match)

Once all of the potential matches have been ranked, the matching records are displayed in a user interface, side by side, so the reviewer can compare the demographic fields and verify if the records are in fact a match. Those with a ranking of 6 are considered to be exact matches and are not usually reviewed. Any matches ranked 1-5 require closer examination by manual review.

Manual Review. Using their domain knowledge registry personnel review the demographic information for more evidence of duplicate records. In addition to an exact match on birth date and highly ranked matches on first and last name, the reviewer attempts to find an exact match on one other demographic field from the following:

- A complete combination of street address, city, state, zip code
- Full telephone number (not including area code only 7 digits)
- Medicaid number
- Social Security Number
- Parent/guardian maiden name if different from the child's last name
- Parent/guardian first name
- Parent/guardian last name if different from the child's last name.

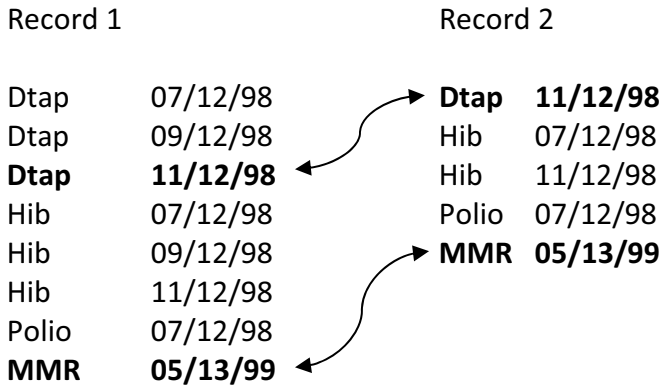
If the reviewer can find an exact match on any one of these demographic items, they can consider these records to be duplicates. Keep in the mind, the reviewer not only sees the most recent information, but all of the records for that particular patient. Therefore, they can use data in earlier records as well as the most recent one to determine a match. This is where linked-based merging is very powerful.

If an exact match cannot be determined, they can also use the following:

- Obvious typos and transposed digits in the street address, but same street name, can be considered duplicates and merged.

- If Alert receives any verbal or written confirmation from a reliable source confirming that the two records represent the same child, consider them duplicates and merge them.
- The reviewer can check with Vital Records to access a birth file for additional information in order to determine if the records are duplicates.

Using Immunization Information. Many times the demographic records are incomplete so there may not be enough information to be assured that the records represent the same person. If the reviewer has strong evidence that the records in question are duplicates, such as exact match on birth date, first and last name, but no other demographic information. They can use immunization information to determine a match. The resolve application can pull up the immunization records associated with each demographic record. Immunization histories can be used if there are at least two pairs of immunizations, each given on a different date and there are no obvious conflicts. Below is an example of two separate matching immunizations each given on the same date.



Notice that there are other possible combinations, but you could not use the hib on 11/12 with the dtap on 11/12 etc., because the dates are the same.

The final matching criteria are for rare cases, to be used only if a match could not be determined using the previous steps.

- If there is only one preexisting record a match can be determined by using only one matching immunization as long as it is not Hepatitis B (HepB).
- Matching based on estimated immunization dates rather than actual immunization dates. This requires knowledge of the immunization record providers. Some providers do not indicate the actual immunization date, but instead put the date that the immunization was processed in their database (e.g. the 30^{th}/31^{st} of the month). So knowing this will allow a match of histories where the dates do not match exactly.
- Match based on two addresses where one is a P.O. Box in the same city or county. However, the addresses must be in a rural county outside Multnomah County, i.e. a low population county

Overview of "Run/Match", the new Alert-IIS Deduplication Process

The Run/Match deduplication algorithm works to minimize duplicate clients in the database by proactively identifying potentially matching records. It uses demographic information to evaluate incoming data against existing records and calculates a composite score indicating how alike the records are. It performs the comparison as records are imported into the system; either interactively, when data is entered into the system using the user interface or automatically during electronic data exchange (batch upload).

The Run/Match demographic deduplication algorithm uses a weighted scoring probabilistic matching approach, which incorporates machine learning techniques to adjust the scores depending on the incoming data. Rather than going through a series of steps, this algorithm uses blocking and scoring. Run/Match computes a potential score for a set of records (a cluster) that could be a match. If more than one incoming record is a match (more than one duplicate in the batch), the last record input will kept as the actual record.

When a match score is high enough to indicate a duplicate record, the current name (first, middle and last) is saved in the "AKA" table. This is done so name changes can be tracked along with the current record. Then the entire record is placed in an archival table, used for research purposes, and the new record overwrites the original data completely. There is only one demographic record per person at anytime in the database. A record from Vital Records is seen as the gold standard and cannot be overwritten by any record not from vital records. See Appendix G for the demographic table schemas.

Identify the initial potential duplicates ("Blocking")

In order for a record to be included in the potential duplicate list, some fields from the incoming record must match exactly to fields in an existing record (a candidate record), in one or more of the following criteria:

- First name with spaces and connector punctuation removed
- Last name with spaces and connector punctuation removed
- Birth date or Death date

Unlike the current Alert process, the new process does not use the IDS field of an incoming record to determine if a pre-existing version of the record is already in the database. In this case the new algorithm misses the opportunity to short circuit the comparison process with a simple, single field, rule based match. However, the new system has been guaranteed by HP, to be more than powerful enough to overcome this difference.

Calculate a potential duplicate score ("Scoring")

Once the candidate list has been selected, a more thorough comparison of the records can take place. Each field of the potential duplicates is compared to the existing record identified in step one. The fields are then scored independently and contribute to the overall composite score. The following fields are used to determine the composite score:

- Name (First, Last, Middle, Suffix)
- Gender
- Birth date
- Death date
- Mother's first name
- Mother's maiden name
- Birth count
- Primary address
- SSN
- Medicaid Number

The individual algorithms used to perform the scoring for each field have been designed (using machine learning techniques) around the domain conventions corresponding to

each data type. For example, first names can be nicknames, shortened names or initials, and last names can be hyphenated and the patient can be referred to by either component of the hyphenated name

Using these techniques, an evaluation of the quality of the match is also made and given a score. The individual field scores reflect how closely (or not) a potential duplicate is in relationship to the existing record. Then each score is weighted and a total score is calculated for the potential match. If the score is at or above the “match” threshold, the record is considered a duplicate. If it is below the "do not match" threshold it is consider a non-duplicate and if it is between the two it is marked for manual review. In many instances the threshold can be configured by the organization. However, the Alert Immunization registry will not have this functionality. Not all of the fields need to be present to achieve a match and not all fields need to match, identically, for an existing record to be considered a match. Individual components (first/middle/last names, house number/street/city/zip) are parsed and scored separately. See Appendix H for examples of Run/Match outcomes.

Records that are marked for manual review are stored separately. An authorized user can then view both the existing record and the new record to determine if they are indeed a match. The goal of this system is too have much less need for manual review than the current Alert deduplication process.

Matching Criteria

Names. These fields provide the most complexity to the overall duplicate detection process. There are numerous issues to be compensated for when attempting to match based on names. People can be known by several different names, sometimes there are first names crossed with last names or middle names. In many instances there are typographic or transposition errors. When the first, middle and last names are evaluated, the system first removes all punctuation and spacing. It then calculates a score for each part of the name.

For all names, the frequency of occurrence of the name in the total dataset is considered, with a more frequent name given a lower score because it has a higher probability in the total population (this makes the name less distinguishing). First names are evaluated for name variants and nicknames (e.g. Michael/Mike/Mikey) and two part names are evaluated separately (e.g. Bobby Joe vs. Bobby). For middle names, the fields are compared but not evaluated for variants, and matches are also made between initials and complete names (e.g. "J" and Jim), but receive less weight. For last names, the evaluation criteria are similar to those for first names with the addition of examining the individual components of compound names separately. For example, Gomez-Rodriquez could match either Gomez or Rodriquez, although these matches receive less weight.

To account for typographic occurrences such as, double to single consonants, abbreviations, like-sounding letter constructions and typographical errors, a combination of string distance and word pattern match algorithms are used. If this technique fails to match, the strings are evaluated phonetically for like sounds. This is achieved by comparing like sounding single or multiple-letter constructions in the word pair as well as using Soundex for phonetic encoding.

After evaluating first and last name, an ethnic name penalty may be applied to the score for a name if it is not common in the population as a whole, but is common as part of an ethnic group. For example, the last name Gupta and the first name Chandra may be rare in the overall population, but they are very common Indian names and are not as selective in that sub population, as the initial name score would suggest.

Birth and Death Dates. The system first attempts to find an exact match for birth or death dates. If it cannot find an exact match on the date then it attempts to find near-date comparisons or typographical errors. Non exact matches are scored lower than an exact match. Some examples of typographical errors include:

- Reversed month and day values (**01/12**/2004 v. **12/01**/2004)
- One invalid component (01/**12**/2004 v. 01/**15**/2004)
- Reversed digits in same field (**01**/12/2004 v. **10**/12/2004)
- Month or year is off by one digit (01/12/**2003** v. 01/12/**2004**)

Address. The address field is parsed into individual elements (e.g. street number, street name, city, state, P.O. Box). Then each individual element is evaluated for exact matches or near-matches. Partial scores are given for matching various components, such as matching street name but with missing number. A partial score is also given for matching city, state and zip, even if the street does not match.

SSN/Medicaid ID Matching. Matches can be made comparing Social Security Numbers and Medicaid Numbers. If an exact match is not found, the program performs a near-match on the field. A score can be assigned when there are up to two differences between the fields.

Scoring

Scoring is done probabilistically such that a close match receives a percentage of the perfect match score. Run/Match assigns a positive, zero or negative score to each field evaluated. A positive score indicates some level of matching, a zero score means evaluation could not be completed (one or both fields are empty or non-specific) and a negative scores indicates the fields are not matched to some degree (e.g. completely different dates, gender values, etc.).

The scores for each field are further divided into three separate groups, which are as follows:

- The family group which consists of the last name, the mother's first name, the mother's maiden name and the parsed address fields.
- The birth event group which consists of the birth date and birth order fields.

- The individual group which consists of the first name, middle name, name suffix, gender and the date of death fields.

The purpose of these three groups is to combine the weights given to individual fields. Grouping by family combines family names with addresses. The birth event group gives strong preference to birthday, but also takes into account birth order as well as multiple births e.g. twins. The individual group narrows the match probability down to each patient. The groups are used as a control measure and keep one set of data types from overly influencing the total match score. This is especially important with siblings, especially twins, since they have so many of the same demographic components.

The family group score is capped at 25 and the totals for all three groups are summed together for the final composite score, which is used for the final evaluation. Only candidates with a composite score at or above the match threshold can be considered an actual match. Table 12 lists the score ranges given to the different fields.

Table 12: Individual Field Scores

Field	Score
Client first name:	-6 to +21
Client middle name:	-3 to +5
Client last name:	-8 to 17
Client suffix:	-10 to 5
Gender:	-7 to +1
Birth date:	-16 to +14
Death date:	-16 to +14
Mother's first name:	-4 to +15
Mother's maiden:	-3 to +15
Primary address:	-2 to 15

Finally, two scoring adjustments can be made by HP clients. The Ethnic Penalty adjustment allows the client to adjust the points given to the potential duplicates' first, last and middle name depending on the registries needs and the size of the ethnic population. There is also a Perfect Name bonus that allows the system to apply bonus points if the first name, last name and date of birth are an exact match. The values, similar to the high and low thresholds, are configurable by the site using an administrative tool.

With all scoring type algorithms there is a balance between the settings for the high and low end of the threshold. The high end of the threshold corresponds to reducing the false positive rate. If it's set too high, many records that should be automatically matched will be selected for manual review. If it's too low, then there could be more false positives. The same goes for the low end, if it's too low, then obvious non duplicates will be submitted for manual review. On going testing will need to be done to ensure that Run/Match is properly matching duplicate records, not overly rejecting duplicates and properly finding non-duplicates.

Side by side look a the Current and New Alert Systems

Data Input Constraints

Both systems have input constraints designed to keep erroneous data from entering the system. The new Alert-IIS utilizes both more stringent input constraints as well as specified code sets which limit the number of fields that must accept "free form" input. This should control the amount of anomalous data that enters the system. Tables 13 and 14 compare the input validation controls between both systems.

Table 13: Current Alert Input Constraints

Alert Input Constraints	Description
Record Identifier	Any record identifier that has previously been submitted will be compared to the current record; if the data is different, the new data will overwrite the previous entries.
Out of range Social Security numbers	1. SSN's consisting of repeating digits are discarded: e.g. 999999999 etc. 2. SSN's which have less than 9 digits are discarded
Non vaccine CPT codes	CPT codes that do not refer to a valid vaccination are discarded.
Improper dates of birth	1. Records with the date of birth before 1900 are discarded. 2. Records with the date of birth after the vaccination date are discarded.

Table 14: New Alert-IIS Input Constraints

Alert IIS Input Constraints	Description
Record Identifier	Demographic record rejected if not specified.
First Name Rules	1. The first name must not be in false names list, e.g., "Baby", "Baby Boy", etc. 2. The first name must be at least two characters in length. If there is no first name "NO FIRST NAME" must in place for the record to be accepted. 3. Must be an alpha (a-z, A-Z) character or a slash, hyphen, period or <space>.
Last Name Rules	1. Must not be in the "false names list", for example: "baby", "baby boy", "no name" etc. 2. Must be at least two characters in length. 3. Must be alpha (a-z or A-Z) characters or a slash, hyphen, period or <space>
Birth Date Rules	1. Cannot be blank 2. Must be in the format "mmddyyyy" 3. Date must be on or before "today's" date 4. Date must be after 01/01/1890

NOTE: Keep in mind that both the current Alert registry and the new Alert-IIS import demographic and immunization information separately. The Record Identifier field is used to link a demographic and immunization record for a single patient.

Table 15: Comparison of Data Integrity Constraints

Location	Problem Type	Current Alert	New Alert-IIS
Field	Illegal values	See Table 12 for input constraints.	See Table 13 for input constraints.
Record type	Uniqueness violation	Most fields are free form entry so this problem has a higher probability of occurrence	Only a few fields allow free form entry e.g. names and parts of the address field.
Record	Referential Integrity Violation	Free form entry allows out of bounds entries for several fields	Predefined code sets for most fields avoid referential integrity violations
Field	Missing values	Has required fields for basic components of a medical record, other fields are allowed to be blank for unknown value	Has required fields for basic components of a medical record, when value is unknown an unknown value is indicated
Field	Embedded values	Free form fields that are expected to have names are parsed for invalid characters	Free form fields that are expected to have names are parsed for invalid characters
Field	Improper values	No address checking algorithms employed	Performs city, state comparison with zip code

Data Quality Controls

The current Alert registry has problems with controlling data quality. Due to the fact that most, if not all, fields are free form input, a great deal of "dirty" data has entered the registry. The new Alert-IIS utilizes integrity constraints along with predefined code sets to prevent dirty data from proliferating within the system. Table 15 compares the schema and instance level constraints.

Deduplication Comparison

The deduplication processes between the current Alert and the new Alert-IIS are very different. The current Alert system relies heavily on manual review to compensate for its data warehouse design; whereas the new Alert-IIS attempts to maintain the

Table 16: Matching Algorithm

Current Alert System	New Alert System
Performs a pair wise comparison of records. For the demographic record, single-field "blocking" on exact match of birth date. Then multi-field phonetic matching on Soundex encoded first and last name. Next steps are human review, it uses a rule-based algorithm on an exact match for one other demographic. Human review uses a clustering approach; all records in a cluster can be used to determine a match.	For the demographic record single-field "blocking" on birth date or name. Then uses probabilistic machine learning techniques based on first and last name, plus ethnicity, gender, address and/or SSN. Then It calculates an overall score for the demographic information. If the score is above the threshold it is a match, if below, it's a non-match. If it is with the range between the two thresholds, it is marked for further review.

Table 17: Deduplication Process

Current Alert System	New Alert System
A combination of Front-end & Back-end matching. Checks to see if the incoming record identifier (IDS) matches a pre-existing record. If there is a match, record is updated with new information if any. New records that do not have an exact record identifier match are filtered prior to adding them to the system. Then the Resolve process examines the new records for possible matches. Those that are clear duplicates are linked under a new master patient index; those that are potential duplicates are submitted for manual review. The Auto-Merge algorithm is run periodically on the database, in batch mode, to look for missed potential duplicates.	The database should exist in a permanently deduplicated state. Incoming records are compared to existing database records. If a match is found, and the new record contains additional or different information; then the previous record is marked as an archival record and the new complete record overwrites It. There is some merging of fields as well. Front-end matching. New records that come in through the user interface are compared to pre-existing records. If those records are a match, the UI presents the match to the user allowing them to either add the new information to the previous record and/or update the pre-existing information if it is incorrect. For electronic data loads, the new record fields are compared to previous records. If there is a match the previous record is marked as historical and the new record becomes the current one. For any borderline potential match, those records are marked for manual review.

information in an error free state by preventing duplicate records from entering the database. Tables 16 and 17 compare the deduplication strategies used by both systems.

Methods: A Comparison of the Deduplication Processes

In order to evaluate the difference in deduplication accuracy between the current and new systems, we processed a single set of demographic records with known duplicates through both deduplication systems. In addition, since the current process is predominantly a manual process, we attempted to quantify the time and effort it took for an experienced staff member to process the test set. The new system should process almost all, if not all, of the records automatically. There is the possibility that some records will be set aside for further review, but we are expecting this number to be small. We will not be able to document the time it takes for someone to manually review those records since that functionality is not currently available. We will attempt to estimate the time using the staff's domain knowledge.

Test Data Set. In order to properly exercise the deduplication algorithms the test set needs to emulate duplication anomalies commonly experienced by immunization registries. A fictitious data set created for the CDC was available for this purpose.[42] It contains 550 total records with 299 unique records and 251 duplicates of those unique records. Each record is a combination of demographic and immunization information for a single child. It has been specifically designed to assist all registries with assessing their system's ability with either preventing or removing duplicate records. The test cases represent various duplicate record anomalies that appear in real immunization data such as the following:

- First & last name spelling errors
- First & last name hyphenation
- First, last and middle names in wrong fields
- First name as "baby", "baby girl" etc
- Date of birth differences
- Gender difference
- Different last name
- Suffix included in first or last name
- Duplicate core data
- Exact duplicate

The test data has a high percentage of duplicate records, 49%, which is higher than what is typical for the average registry. According to the CDC, most registries report that about 10% of their records are duplicates. At Alert, that number is not tracked, so no comparison is available.

Along with the duplicate anomalies, the test data has a number of non-duplicate records that could be mistaken for duplicates. Some test cases are unique records with similarities to other records in fields such as last name, mothers name and/or birth date, where in fact these could be records from twins, siblings or similar by random chance. These records have the potential to demonstrate where a process may mismatch non-duplicate records. See Appendix I for more information regarding the duplicate and non-duplicate record types.

The current and new Alert systems can also examine other fields not present in this test data set, including home address, home telephone, Social Security and Medicaid number. However, since the all of the fields in the test records are used for discovering duplicates in both systems, the duplicates should be discovered and deduplicated. See Examples 1 & 2 for the fields present in the test set.

Finally, records that are rejected for various reasons such as improper first names, e.g. 'baby boy', or impossible birthdates are deemed to be the equivalent of reducing the number of duplicates.

Example 1: Example Demographic Record

Last Name	First Name	Middle Name	Suffix	DOB	Sex	Mom Maiden	Mom Last	Mom First	Mom Middle
CALDWELL	BRITTANY	ABIGAIL		19970224	F	PARKER	CALDWELL	IRENA	LORA

Example 2: Example Vaccination Record

VacName	VacCode	VacMfr	VacDate
HIB-HbOC	47	WAL	19970404

Test Methods. The test set was formatted and then processed through both deduplication systems. Once both systems finished processing the test set, all of the records were extracted and analyzed using an evaluation tool provided by the CDC that is designed to quantify how well the system performed at deduplicating the test data.

Each system can be given two scores, one for how well the system performed at finding individual duplicate records out of all duplicate records in the test set, i.e. the sensitivity score; the second for how well the system performed at avoiding labeling non-duplicates as duplicates, avoiding false positives, i.e. the specificity score. A sensitivity score of 100% would be equal to finding all of the duplicates in the test set, regardless of how many non-duplicates were mislabeled as duplicates. A specificity score of 100% would be equal to not mislabeling any non-duplicates as duplicates. Based on tests performed by the CDC, a sensitivity score of 90% or better combined with a specificity score of 98% or better has been recommended as the minimum goal for immunization registries.

The test method for the *current* system was performed in two stages evaluating the semi-automated stage separately from the manual review stage in order to determine how accurate and time consuming each stage is. Records were uploaded into the database as new records, each with a single MPI and the test was started using Auto-Merge; we did not have access to processing the records through the Import/Export system. Auto-Merge processed the records using the algorithm described earlier and performed the initial deduplication step of ranking all of the potential duplicates. We then took all of the level 6 ranked matches and deduplicated them using the MPI. At this point the test set was extracted and analyzed for Stage 1. Next, we continued with manual review from the point where the Auto-Merge stage left off. This is how the process is normally performed by registry staff. The entire partially deduplicated test set was extracted into a temporary table and an experienced Registry staff member

performed manual deduplication of the data. Finally, the test set was again extracted and analyzed for Stage 2.

The test method for the *new* system could only be done in a single stage. The test data was formatted to the specifications of the new system and placed in a fixed field length text file. The file was then copied to a specific folder location on the new system. A batch process was started on the new system which imported the text file and preprocessed it for input constraints and validation errors. Finally, the system processed each record in the file through Run/match and either loaded the record or handled the record as a duplicate. When it finished processing the test set, we extracted all of the records for evaluation and examined any records that were marked as potential duplicates. This is equivalent to Stage 1 in the current system evaluation. A Stage 2 dataset could not be obtained for the new system.

Results

Current Alert System – Stage 1 Auto-Merge

A total of 550 records, 299 non-duplicates and 251 duplicates were processed by the system. Resulting from this, 421 records were added to the database as individual records, including 123 undetected duplicates and 298 non-duplicates. 128 duplicate records were merged and 1 non-duplicate record was mislabeled as a duplicate. See table 18 for a results summary.

The automated part of the current deduplication system had a 51% sensitivity rate and a 99.67% specificity rate. According to the CDC's recommendations, the sensitivity rate is 39% below the minimum sensitivity level of 90%. While the specificity rate is 1.67% above the minimum recommended specificity level of 98%. It took approximately 1.5 hours to complete the Auto-Merge process. This includes starting the application, uploading the data, waiting for the application to create the potential duplicate list, saving the data as a file, running another sub-process on that file to perform the ranking and finally

Table 18: Results after Auto-Merge, Prior to Manual Review

Summary of Results	Records
Actual Total Records in Test Data	550
Total Duplicates in Test Data (2 records each)	251
Total Non-duplicates in Test Data	48
TEST RESULTS:	
Total New Clients Added to Registry (Unique IDs)	421
Undetected duplicates added as new records	123
Duplicate records merged, rejected, or flagged as duplicates	128
Records handled as certain duplicates (merged with match or rejected)	128
Records handled as potential duplicates (flagged)	0
Missing records (mistaken for duplicates or rejected)	1
Records correctly identified as non-duplicates	298
Non-duplicates flagged as possible duplicates	0
Overall Score for Duplicate Record Detection (Sensitivity)	51%
Overall Score for Accuracy in Duplicate Record Determination (Specificity)	99.67 %

saving the level 6 ranks as duplicate records. The application crashed once during this process.

Current Alert System – Stage 2 Manual-review

Continuing from stage 1, a total of 550 records, 298 non-duplicates and 123 undetected duplicates and 129 deduplicated records were examined during manual review. Resulting from this, 338 records were added to the database as individual records; this includes 40 undetected duplicates and 298 non-duplicates. 211 duplicate records were merged and 1 non-duplicate record was incorrectly labeled as a duplicate. . The manual part of the current deduplication system had an 84.1% sensitivity rate and a 99.67% specificity rate. According to the CDC's recommendations, the sensitivity rate is 6.6% below the minimum sensitivity level of 90%. See table 19 for a results summary.

The manual review process increased the sensitivity score by 33% to 84.06%, but it took approximately 2 hours for an experienced registry staff member to deduplicate the remaining record set. This includes the fact that 123 of the duplicate records had

Table 19: Results after Auto-Merge and Manual Review

Summary of Results	Records
Actual Total Records in Test Data	550
Total Duplicates in Test Data (2 records each)	251
Total Non-duplicates in Test Data	48
TEST RESULTS:	
Total New Clients Added to Registry (Unique IDs)	338
Undetected duplicates added as new records	40
Duplicate records merged, rejected, or flagged as duplicates	211
Records handled as certain duplicates (merged with match or rejected)	211
Records handled as potential duplicates (flagged)	0
Missing records (mistaken for duplicates or rejected)	1
Records correctly identified as non-duplicates	298
Non-duplicates flagged as possible duplicates	0
Overall Score for Duplicate Record Detection (Sensitivity)	84.06%
Overall Score for Accuracy in Duplicate Record Determination (Specificity)	99.67 %

already been discovered and marked as duplicates. They also failed to find the non-duplicate which was incorrectly marked as a duplicate record.

A further analysis of the duplicate records demonstrates areas of weakness for the current process. When examining the missed duplicate records list, the largest group missed was the “date of birth difference” group. This group made up 75% of the total missed duplicates. First name spelling errors combined with first, middle and last names in the wrong fields made up the other 25%. See table 20 for duplicate record results.

New Automated Alert System

A total of 550 records, 299 non-duplicates and 251 duplicates were processed by the system. As a result of this process, 310 records were added to the database as individual records, including 15 undetected duplicates, 291 non-duplicates and 6 mis-merged non-duplicates. 234 duplicate records were merged, 2 duplicates were flagged as potential duplicates and 2 non duplicate records were lost. See table 21 for a results summary.

Table 20: Results for Duplicate Records found with the current system

Results for Duplicate Records					
Problem Type	**Data Total**	**Rejected or Merged**	**Flagged**	**Missed**	**Score**
First Name Spelling	51	46	0	5	90.2%
Last Name Spelling	24	24	0	0	100%
First Name Hyphenation	15	15	0	0	100%
Last Name Hyphenation	23	23	0	0	100%
First Name Reversed w/Last Name	4	4	0	0	100%
First Name Reversed w/Middle Name	4	2	0	2	50%
Middle Name Reversed w/Last Name	4	3	0	1	75%
Different Last Name	14	12	0	2	85.71%
First Name as 'Baby', 'Baby	9	9	0	0	100%
Suffix in First Name	7	7	0	0	100%
Suffix in Last Name	5	5	0	0	100%
Date of Birth Difference	61	31	0	30	50.82%
Gender Difference	4	4	0	0	100%
Duplicate Core Fields	16	16	0	0	100%
All demographic fields identical	10	10	0	0	100%
TOTAL DUPLICATES	251	211	0	40	84.06%

The new automated deduplication system had a sensitivity rate of 93.2% which is 3.2% higher than the 90% recommended by the CDC and 9.1% higher than the current process of 84.1%. It also had an accuracy rate of 97.3% which is .7% below the 98% recommended by the CDC and 2.4% below the current manual process of 99.7%. The two non-duplicate records that were lost were discarded during the input validation process because they had the name "baby" in the first name field. Two duplicate records were flagged as potential duplicates and placed in the pending queue. These records would require some manual review to confirm whether they were in fact duplicates. This process can be estimated to take about 2-4 minutes for an experienced person. The new deduplication process took a mere 12 seconds as compared to 3.5 hours for the current deduplication process.

A further analysis of the results for the 15 duplicate records that were missed (false negatives) and the 6 non-duplicate records that were mismatched and merged (false positives) was performed.

Table 21: Results Summary for the Automated Deduplication Process

Summary of Results	Records
Actual Total Records in Test Data	550
Total Duplicates in Test Data	251
Total Non-duplicates in Test Data	299
TEST RESULTS:	
Total New Clients Added to Registry (Unique IDs)	310
Undetected duplicates added as new records	15
Duplicate records merged, rejected, or flagged as duplicates	236
Records handled as certain duplicates (merged with match or rejected)	234
Records handled as potential duplicates (flagged)	2
Missing non-duplicate records (rejected)	2
Records correctly identified as non-duplicates	291
Non-duplicates mis-merged as duplicates	6
Overall Score for Duplicate Record Detection (Sensitivity)	93.23%
Overall Score for Accuracy in Duplicate Record Determination (Specificity)	97.25%

Example 3: First Name Problem

ID	Lname	Fname	Mname	DoB	Sex	MoLname	MoFname	MoMname
54359	ARRON	HENRY	JAMES	19971213	M	ARRON	CLARA	JOSEPHINE
54328	ARRON	**BOY**		19971213	M	ARRON	CLARA	JOSEPHINE

False negative analysis (missed duplicate records)

For each example of missed duplicate records, both records are shown. However, according to the evaluation tool, only one record of each of these pairs should have been kept.

First names as 'baby', 'baby girl', etc. In example 3, records with a first name of "Boy" should have been prevented from entering the system. Alert may want to have HP add "BOY" to the false names list.

Date of birth difference. In example 4, three pairs of records with date of birth differences avoided being deduplicated by the system. Some records had other fields that were also not an exact match. The difference between the birthdates gives an indication of the selection criteria used by the Run/Match scoring algorithm.

Example 4: Date of Birth Differences

ID	Lname	Fname	Mname	DoB	Sex	MoLname	MoFname	MoMname
54480	LOTT	MARK	ANTHONY	**19970530**	M	LOTT	BARBARA	
54440	LOTT	MART	ANTHONY	**19970510**	M	LOTT	BARB	
55999	PATEL	JYOTI		**19960304**	F	PATEL	URVI	NAYNA
56060	PATEL	JYOTI		**19960804**	F	PATEL	URVI	NAYNA
56053	DIAZ	KRISTY		**19970201**	F	DIAZ	ROSAMARIA	
56055	DIAZ	KRISTY		**19970102**	F	DIAZ	ROSA	MARIA

Example 5: Last Name Hyphenation Difference: a duplicated record not identified

ID	Lname	Fname	Mname	DoB	Sex	MoLname	MoFname	MoMname
54426	**NAMYNANIK**	**ARVEE**		19960604	F	**NAMYNANIK**	MEENU	SAMIR
54417	NAMYNA-NIK	**NIK**	**ARVEE**	19960604	F	**NAMYNA**	MEENU	SAMIR

Last name hyphenation. In example 5,. most records with last name hyphenation discrepancies where handled correctly. One special case had many different issues, e.g. child and mother's name differences.

Middle name and last name reversed. In example 6, the record had middle name reversed with last name, a different birth date and the mother's first name was different. An argument could be made that these two records should be seen as unique records rather than as duplicates.

Example 6: Middle and Last Names Reversed: a duplicated record not identified

ID	Lname	Fname	Mname	DoB	Sex	MoLname	MoFname	MoMname
54278	TAYLOR	MADISON	MORGAN	19961127	F	MORGAN	SHANNON	MELISSA
54414	MORGAN	MADISON	TAYLOR	19970118	F	MORGAN	KIMBERLY	

Records marked as potential duplicates. In example 7, the two pairs of records were marked as potential duplicates. The second record of these pairs was placed in the pending queue. A manual inspection of the records would be required to determine if they were duplicates or not.

Example 7: Marked as Potential Duplicates

ID	Lname	Fname	Mname	DoB	Sex	MoLname	MoFname	MoMname
54453	BATES	EMILY	CAMILLE	18980504	F	BATES	BLAIR	ELLEN
55976	BATES	EMILY	CAMILLE	19990504	F	BATES	BLAIR	ELLEN
54339	NAQUI	SAYED		19970324	F	NAQUI	SURAKAK	HEMA
55995	NAQVI	SIED		19970324	F	NAQVI	SURAKAK	HEMA

Example 8: First Name Spelling Differences

ID	Lname	Fname	Mname	DoB	Sex	MoLname	MoFname	MoMname
54436	ANDERSON	GINGER	ROBERTA	19960727	F	ANDERSON	CLEO	URSULA
54475	ANDERSON	GINNY	ROBERTA	19960727	F	ANDERSON	CLEO	URSULA
54458	NOONAN	ANDY		19980202	M	NOONAN	DEBORAH	ELIZABETH
56021	NOONAN	DREW		19980202	M	NOONAN	DEBORAH	BETH
54464	CHANG	YANG	SOOK	19960302	M	CHANG	JUNG	HYUM
56013	CHANG	YOUNG	SOOK	19960302	M	CHANG	JUNG	HYUM
55977	HAYAKAWA	ATSUSHI		19960609	F	MASARU	EYUKA	
54335	HAYAKAWA	SUSIE		19960609	F	MASARU	EYUKA	
56007	ACOSTA	STEPHANIE	SUZANNE	19961012	F	CRUZ	ANGELICA	MARIA
56056	ACOSTA	SUSAN	SUZANNE	19961012	F	CRUZ	ANGIE	MARIA
54306	PATEL	DHILIP		19970708	M	PATEL	VARSHA	MANJULA
56035	PATEL	PHILIP		19970708	M	PATEL	VARSHA	MANJULA
56065	DAWSON	ANTHONY		19961031	M	DAWSON	RACHEL	H
55989	DAWSON	CLAYTON		19961031	M	DAWSON	RACHEL	H
54281	NGUYEN	SAN		19970724	M	NGUYEN	TRINH	VAN
54347	NGUYEN	SON		19970724	M	NGUYEN	TRINH	VAN

First name spelling errors. Example 8, shows eight pairs of records that have first name errors and were missed by the deduplication algorithm. In several of these cases the first names would have very different phonetic encodings.

It can be argued that some of these undetected duplicates are acceptable, given the fact that it's extremely important to avoid false positives, especially when it comes to handling twins or siblings. Plus, in all of these examples, there is a limited amount of information. With other data points, such as address, phone, Social Security and Medicaid numbers there is a good chance Run/Match would be able to make a better automated determination. When using an automated system like Run/Match there is always some trade off between sensitivity and specificity.

False positive analysis (non-duplicates incorrectly merged)

Incorrectly deduplicated records had individual fields merged into what was considered to be their duplicate records with loss of information. The CDC evaluation tool determined these to be false positives. However, with a closer examination, one could argue that Run/Match performed the correct operation for several of these record pairs. In addition, these records had the immunization information from both records linked to a single record. In all of the cases below, the record with the ID number is the resultant record that was kept in the database. Again, the test data has limited information, and with other data points such as address, phone, Social Security and Medicaid numbers there is a good chance Run/Match should be able to make a better automated determination. In addition, any record that received its first name, last name and date of birth values from Vital Records cannot have those fields overwritten. These cases highlight the potential drawback of having a deduplication system that overwrites current information with new information if that new information is incorrect. The underline in the following examples indicates the information that was lost in the final record. The immunizations are linked to the record with the ID number.

Same birth date but last names do not match. Example 9 shows two records with different child's and mother's last name, but same date of birth. This could occur, for example, if a mother remarried. However, these two records were meant to be two different children with very similar demographic information.

Example 9: Same Birth Date, Last Names do not Match

ID	Lname	Fname	Mname	DoB	Sex	MoMaiden	MoLname	MoFname	MoMname
54284	PERRY	JESSICA	LESLIE	19960407	F	LLOYD	CAMPBELL	KATHERINE	FAY
First	PERRY	JESSICA	LESLIE	19960407	F	LLOYD	PERRY	KATHY	FAY
Second	CAMPBELL	JESSICA	LESLIE	19960407	F	LLOYD	CAMPBELL	KATHERINE	FAY

Example 10: Same First & Last Name but Different Birth Date

ID	Lname	Fname	Mname	DoB	Sex	MoMaiden	MoLname	MoFname	MoMname
55971	CRATER	JASON	ALEXANDER	19960715	M	TANNER	CRATER	CLAIRE	
First	CRATER	JASON	ALEXANDER	19960714	M	TANNER	CRATER	CLAIRE	
Second	CRATER JR	JASON	ALEXANDER	19960715	M	TANNER	CRATER	CLAIRE	

Same first and last names and mother's first and last names. Example 10 is an example of merging. Run/Match chose the record with the birth date on the 15th, but merged the first name from the record with the birth date on the 14th. Plus, one could make an argument that these records are indeed duplicates that have a typo in the birth date, even though the evaluation tool indicated these were false positives.

Same birth date, first name, middle name, mothers maiden and first name. Example 11 shows what can happen with incorrect merging. Notice how the combined record has the last name of HAYES for the mother vs. the child's last name of HAYS.

Example 11: Same Birth Date, Different Names

ID	Lname	Fname	Mname	DoB	Sex	MoMaiden	MoLname	MoFname	MoMname
54288	HAYS	CHRISTOPHER	ERIC	19970208	M	TUCKER	HAYES	LAURA	
First	HAYES	CHRISTOPHER		19970208	M	TUCKER	HAYES	LAURA	
Second	HAYS	CHRISTOPHER	ERIC	19970208	M	TUCKER	HAYS	LORI	

Same birth date, last name, mother's last name. Example 12, is a good example of non-duplicates which can appear to be duplicates. Notice that the merged record doesn't represent either child correctly. The child's first and last name from one record is combined with the mother's first and maiden name from the other.

Example 12: Same Birth Date, Last Name, Different First Name

ID	Lname	Fname	Mname	DoB	Sex	MoMaiden	MoLname	MoFname	MoMname
54483	WOOD	NICHOLAS	JOHN	19990912	M	COLEMAN	WOOD	SUSAN	
First	WOOD	JOHN	NICHOLAS	19990912	M	COLEMAN	WOOD	SUSAN	
Second	WOOD	NICHOLAS	JOHN	19990912	M	BAILEY	WOOD	KIMBERLY	ANN

Example 13: Same Birth Date & Last Name, Different First Name

ID	Lname	Fname	Mname	DoB	Sex	MoMaiden	MoLname	MoFname	MoMname
54385	MOON	JIN		19990502	M	LEE	MOON	YON-MI	
First	MOON	JIN		19990502	M	PARK	MOON	MIYON	
Second	MOON	SOON-JIN		19990502	M	LEE	MOON	YON-MI	

Same last name, birth date, mother's last name. Example 13 is another example of similar non-duplicates getting merged. Notice that the child's first name from the first record is kept with the mother's first and maiden name from the second record.

Same first, last names and mother's first, last, middle names. Example 14 demonstrates how difficult a duplication determination can be. Even though the CDC evaluation tool indicates these are separate children, the records could easily be considered potential duplicates with a one day difference in the birth dates.

Example 14: Same Last Name, Different Birth Date, First Name

ID	Lname	Fname	Mname	DoB	Sex	MoMaiden	MoLname	MoFname	MoMname
54413	GREEN	JOSHUA	ROBERT	19990824	M	JOHNSON	GREEN	MARY	CYNTHIA
First	GREEN	JOSHUA	DAVID	19990823	M		GREEN	MARY	ANN
Second	GREEN	JOSHUA	ROBERT	19990824	M	JOHNSON	GREEN	MARY	CYNTHIA

Conclusion

The staff at the Oregon Alert Immunization Registry has done a remarkable job at meeting the CDC's goal of a 90+% deduplication rate with a 98+% accuracy rate. This is in spite of the fact that they have a deduplication system that has become over-burdened from an ever-increasing amount of information. The current deduplication process does accomplish the task of achieving a high accuracy rate of 98+%, but this has been achieved at the cost of staff resources needed to compensate for lower data quality and an over abundance of duplicate data within the system.

The *current* deduplication process could be improved in two major areas. First, the system should have stronger data quality constraints. At the schema level, this is accomplished through better database design, employing predefined code sets and minimizing free text fields. At the instance level, this is accomplished by implementing stronger validation controls during data input. The automated deduplication process needs to be improved to achieve a sensitivity rate closer to 90%. Data quality could also be improved by moving away from the all inclusive data warehouse design to one where the database was maintained in a deduplicated state.

Second, the current system could be improved by standardizing the deduplication as well as the data extraction processes. Numerous workflows are followed to achieve deduplication. Opportunities are missed to save the time and effort of repeated duplicate processing. In addition, with people performing different processes, the chance for human error also increases.

The *new* system corrects both of the problems listed above. At the time of this writing, the implementation was still only in the testing phase and we could not compare any workflow processes based on the new systems functionality. Our tests of the deduplication process found that the new system did a remarkable job of finding

duplicates with 93% sensitivity and was highly accurate with 97.25% specificity. This specificity rate is below the 98% recommended by the CDC and it is also below what the Alert registry prefers, which is 100%. However, there is a trade off between effective automation of deduplication rate and accuracy. In addition, the test set is missing several fields used to compute the potential match score and with further examination it could be argued that Run/Match made the correct decision regarding some of the non-duplicates. This would have yielded a score above the 98% recommend minimum. A more complete set of records should be designed to determine a more precise accuracy rate. On the whole, the deduplication performance of the new system is superior to the current process. The tradeoff in loss of specificity of 2.4% is far outweighed by the decrease in time for processing from 3.5 hours to 12 seconds.

Creating a workflow for monitoring the deduplication process by Alert staff is suggested, with a report generated for every batch upload of data. In order for the Alert staff to gain confidence in their new system, someone should review this report daily and follow up on the records that were rejected. They should also consider auditing the system periodically by checking for duplication anomalies. They can approach this by creating a test set and running it through the system and/or by performing a statistical analysis on a random sample of records in the database.

The staff at the registry has done a great deal of work preparing for the conversion to the new system, especially with the resources they have available. The change will not be without its challenges, but they are well prepared for the transition.

[1] Centers for Disease Control and Prevention (CDC). Immunization information systems progress--United States, 2006. MMWR Morb.Mortal.Wkly.Rep. 2008 Mar 21;57(11):289-291.

[2] Rask KJ, Wells KJ, Kohler SA, Rust CT, Cangialose CB. The cost to providers of participating in an immunization registry. Am.J.Prev.Med. 2000 Aug;19(2):99-103.

[3] Government Healthcare BPO Services: Business Case | hp.com Available at: http://h10134.www1.hp.com/services/governmenthealthcare/. Accessed 3/30/2010, 2010.

[4] Canavan BC, Kurilo M, Moss T, McLaren R, Berry K, Thomas C, et al. Immunization information systems progress - United States, 2005. MMWR Morb.Mortal.Wkly.Rep. 2006 Dec 15;55(49):1327-1329.

[5] Orenstein WA, Cordero JF, Willis BM. Childhood immunization registries. JAMA 1996 May 1;275(17):1312-1313.

[6] Hinman AR, Urquhart GA, Strikas RA, National Vaccine Advisory Committee. Immunization information systems: National Vaccine Advisory Committee progress report, 2007. J.Public Health Manag.Pract. 2007 Nov-Dec;13(6):553-558.

[7] Bartlett DL, Washington ML, Bryant A, Thurston N, Perfili CA. Cost Savings Associated With Using Immunization Information Systems for Vaccines for Children Administrative Tasks. Journal of Public Health Management & Practice 2007 November/December;13(6):559-566.

[8] Urquhart G, Williams W, Tobias J, Welch FJ. Immunization Information Systems use during a public health emergency in the United States. J Public Health Manag Pract. 2007;13(5):481–485.

[9] Boyd TD, Linkins RW, Mason K, Bulim I, Lemke B. Assessing immunization registry data completeness in Bexar County, Texas. Am.J.Prev.Med. 2002 Apr;22(3):184-187.

[10] Glazner JE, Beaty BL, Pearson KA, Elaine Lowery N, Berman S. Using an immunization registry: effect on practice costs and time. Ambul.Pediatr. 2004 Jan-Feb;4(1):34-40.

[11] Linkins RW. Immunization registries: progress and challenges in reaching the 2010 national objective. J.Public Health Manag.Pract. 2001 Nov;7(6):67-74.

[12] M. Fowler, G. Simpson and K. Schoendorf , Families on the move and children's healthcare. *Pediatrics* 91 (1993), pp. 934–940.

[13] Fairbrother G, Kuttner H, Miller W, Hogan R, McPhillips H, Johnson KA, et al. Findings from case studies of state and local immunization programs. Am.J.Prev.Med. 2000 Oct;19(3 Suppl):54-77.

[14] Christakis DA, Stewart L, Bibus D, Stout JW, Zerr DM, MacDonald JK, et al. Providers' perceptions of an immunization registry. Am.J.Prev.Med. 1999 Aug;17(2):147-150.

[15] Clark SJ, Cowan AE, Bartlett DL. Private provider participation in statewide immunization registries. BMC Public Health 2006 Feb 15;6:33.

[16] Wikipedia contributors. Data cleansing. Available at: http://en.wikipedia.org/wiki/Data_cleansing. Accessed 04/02, 2010.

[17] Wikipedia contributors. Data dictionary. Available at: http://en.wikipedia.org/wiki/Data_dictionary. Accessed 03/30, 2010.

[18] Wikipedia contributors. Database schema. Available at: http://en.wikipedia.org/wiki/Database_schema. Accessed 03/30, 2010.

[19] Rahm E, Hong HD. Data Cleaning: Problems and Current Approaches. Available at: http://wwwiti.cs.uni-magdeburg.de/iti_db/lehre/dw/paper/data_cleaning.pdf. Accessed 03/30, 2010:3.

[20] Papadouka V, Schaeffer P, Metroka A, Borthwick A, Tehranifar P, Leighton J, et al. Integrating the New York citywide immunization registry and the childhood blood lead registry. J.Public Health Manag.Pract. 2004 Nov;Suppl:S72-80.

[21] Data validation - Wikipedia, the free encyclopedia Available at: http://en.wikipedia.org/wiki/Data_validation. Accessed 3/24/2010, 2010.

[22] Muller H, Freytag JC. Problems, Methods, and Challenges in Comprehensive Data Cleansing. 2003; Available at: http://www.dbis.informatik.hu-berlin.de/fileadmin/research/papers/techreports/2003-hub_ib_164-mueller.pdf. Accessed 03/30, 2010.

[23] Miller PL, Frawley SJ, Sayward FG. IMM/Scrub: a domain-specific tool for the deduplication of vaccination history records in childhood immunization registries. Comput.Biomed.Res. 2000 Apr;33(2):126-143.

[24] Bose SSD, Wheeler W. Verification and validation procedures for immunization registries. Am.J.Prev.Med. 1997;13(2 SUPPL.):62-65.

[25] Davidson AJ, Melinkovich P, Beatty BL, Chandramouli V, Hambidge SJ, Phibbs SL, et al. Immunization registry accuracy: Improvement with progressive clinical application. Am.J.Prev.Med. 2003 4;24(3):276-280.

[26] Salkowitz S, Clyde S, Wild E. Deduplication Technology and Practices for Immunization Registries as a component of Integrated Child-Health Information Systems. Available at :http://www.phii.org/resources/doc_details.asp?id=134. Accessed 3/23/2010, 2010.

[27] Borthwick A PhD. Data Quality The Foundation of an Enterprise Architecture. Available at: http://www.nysforum.org/documents/pdf/et-6-21-05/ChoiceMakerTechnologies.pdf. Accessed 3/23/2010, 2010.

[28] Helena Galhardas, Daniel Florescu, Dennis Shasha, Eric Simon. An Extensible Framework for Data Cleaning. Available at: http://eprints.kfupm.edu.sa/24824/1/24824.pdf. Accessed 3/24/2010, 2010.

[29] Wikipedia contributors. Database normalization. Available at: http://en.wikipedia.org/wiki/Database_normalization. Accessed 03/26, 2010.

[30] Green S, Lutz R. Measuring phonological similarity: The case of personal names. Available at: http://www.lasinc.com/nameinfor/wp_lsa.htm. Accessed 03/26, 2010.

[31] LingPipe: String Comparison and String Distance Tutorial Available at: http://alias-i.com/lingpipe/demos/tutorial/stringCompare/read-me.html. Accessed 3/26/2010, 2010.

[32] Wikipedia contributors. Soundex. Available at: http://en.wikipedia.org/wiki/Soundex. Accessed 03/26/2010, 2010.

[33] Miller PL, Frawley SJ, Sayward FG. Exploring the utility of demographic data and vaccination history data in the deduplication of immunization registry patient records. J.Biomed.Inform. 2001 Feb;34(1):37-50.

[34] Hylton J.A. Identifying Related Records. Available at: http://ltt-www.lcs.mit.edu/ltt-www/People/jeremy/thesis/node6.html. Accessed 3/28/2010, 2010.

[35] Monge AE, Elkan CP. The Field Matching Problem: Algorithms and Applications. Available at: https://www.aaai.org/Papers/KDD/1996/KDD96-044.pdf. Accessed 3/28/2010, 2010.

[36] Monge AE. Matching Algorithms within a Duplicate Detection System. Available at: http://www.sigmod.org/disc/disc01/out/websites/deb_december/monge.pdf. Accessed 3/28/2010, 2010.

[37] Fellegi IP, Sunter AB. A Theory for Record Linkage Journal of the American Statistical Association 1969;64(328):1183.

[38] Berlin J, Motro A. Database Schema Matching Using Machine Learning with Feature Selection. Available at:

http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.11.6470&rep=rep1&type=pdf. Accessed 3/28/2010, 2010.

[39] Borthwick A, Buechi M, Goldberg A. Key Concepts in the ChoiceMaker 2 Record Matching System. Available at: http://dc-pubs.dbs.uni-leipzig.de/files/Borthwick2003KeyConceptsintheChoiceMaker.pdf. Accessed 03/26/2010, 2010.

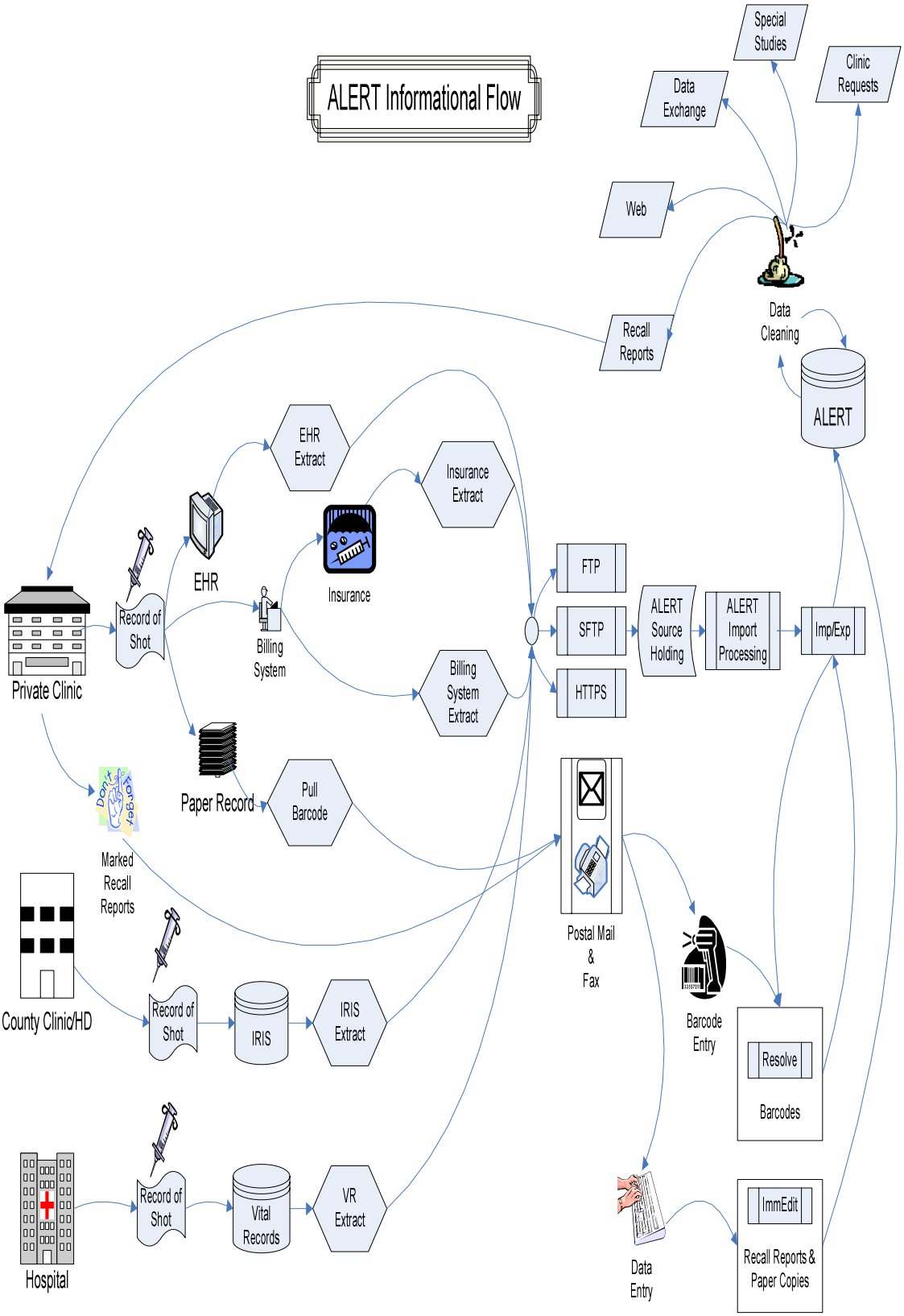
[40] Salkowitz S, Clyde S. De-duplication Technology and Practices for Integrated Child-Health Information Systems. 2003; Available at: http://hln.biz/assets/pdf/dedupe.pdf. Accessed 3/23/2010, 2010.

[41] Wikipedia contributors. Record linkage. Available at: http://en.wikipedia.org/wiki/Record_linkage, Accessed 03/26/2010, 2010.

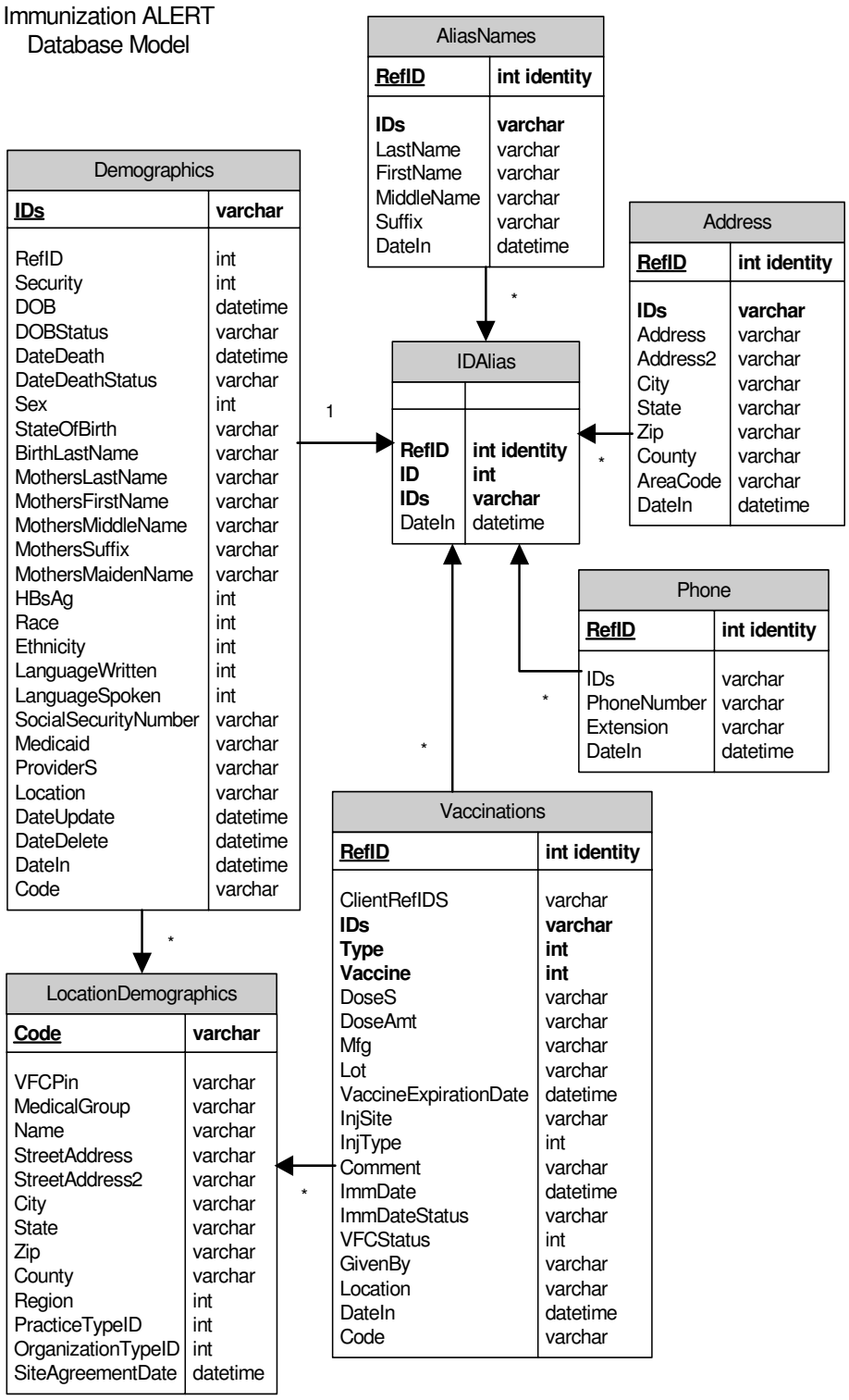
[42] Vaccines: IIS/Tech/Deduplication Test Cases Available at: http://www.cdc.gov/vaccines/programs/IIS/tech/dedup.htm. Accessed 3/20/2010, 2010.

[43] Drawings provided by registry staff. Authors include Amber Crandell and Terry Locke. Accessed 4/20/2010.

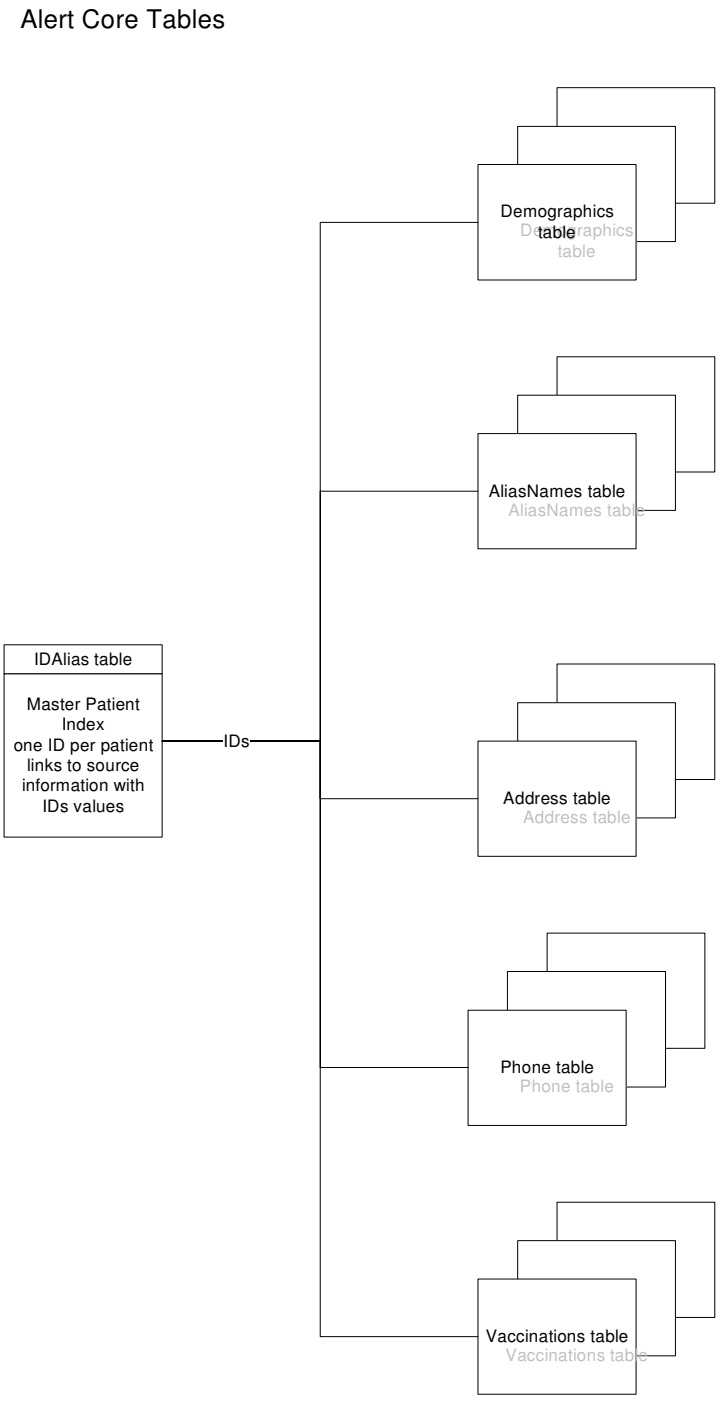
Appendix A: Output & ETL Informational Flow Chart[43]



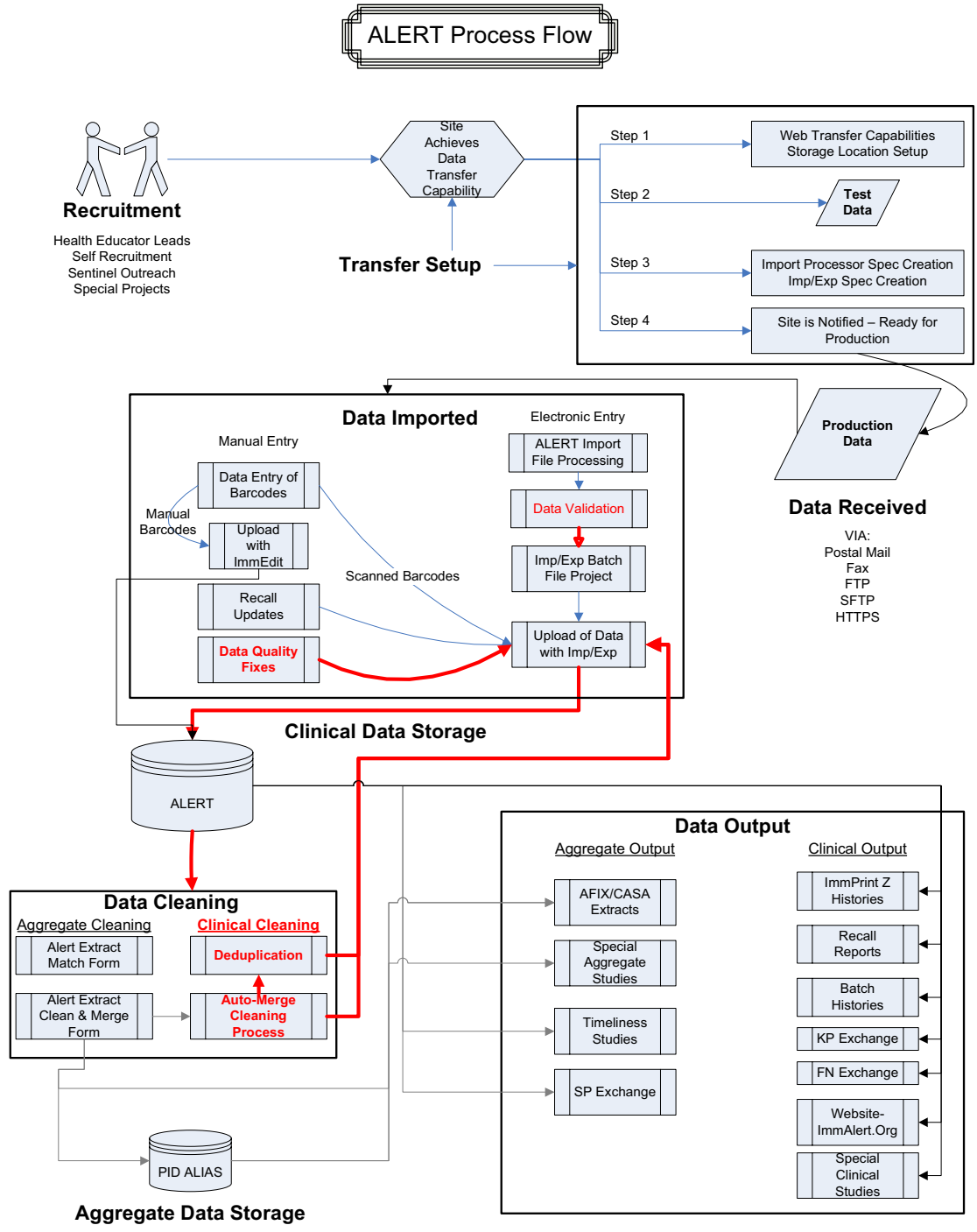
Appendix B: Current Alert Database Schema[43]



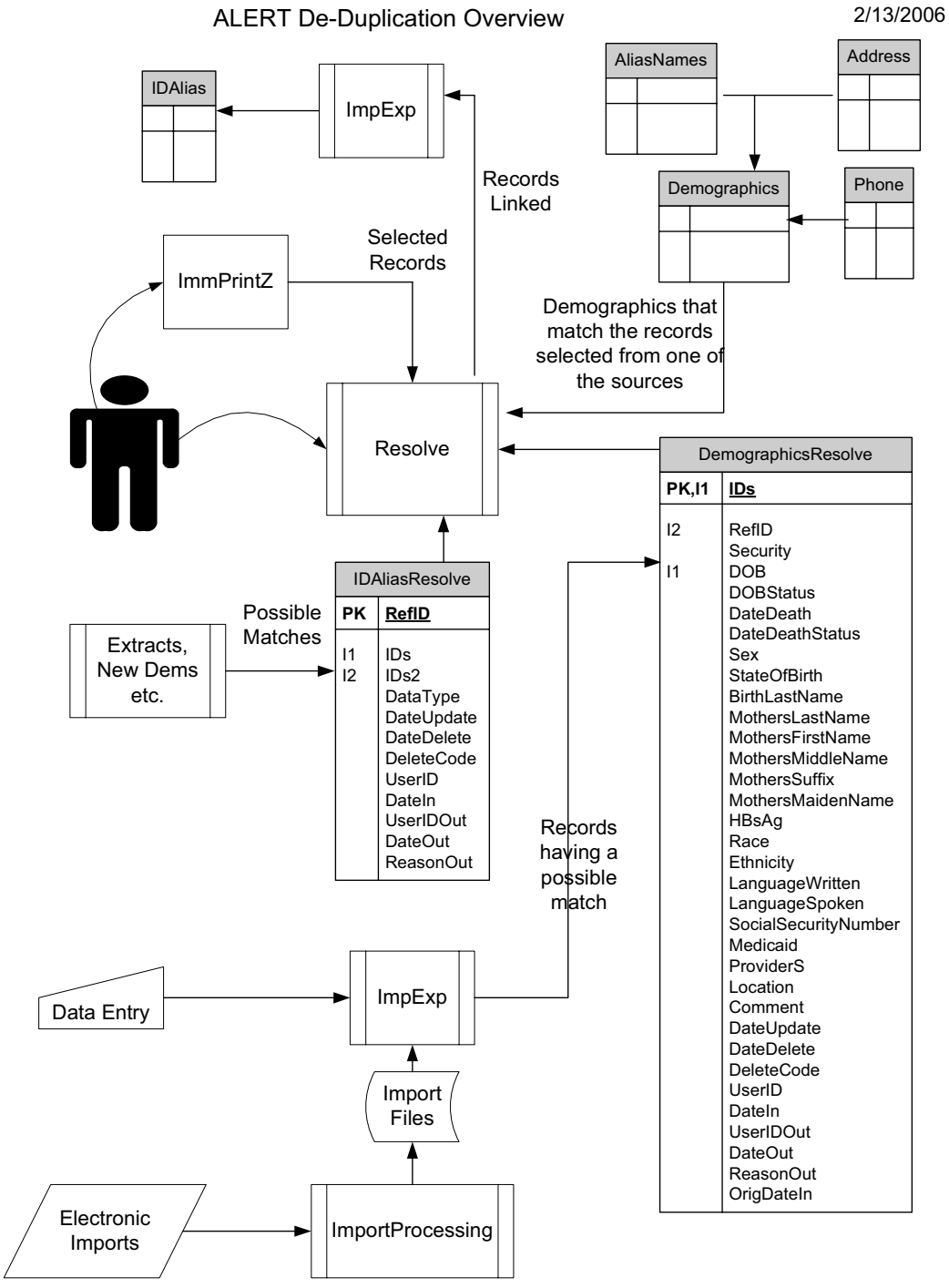
Appendix C: Current Alert Master Patient Index Configuration[43]



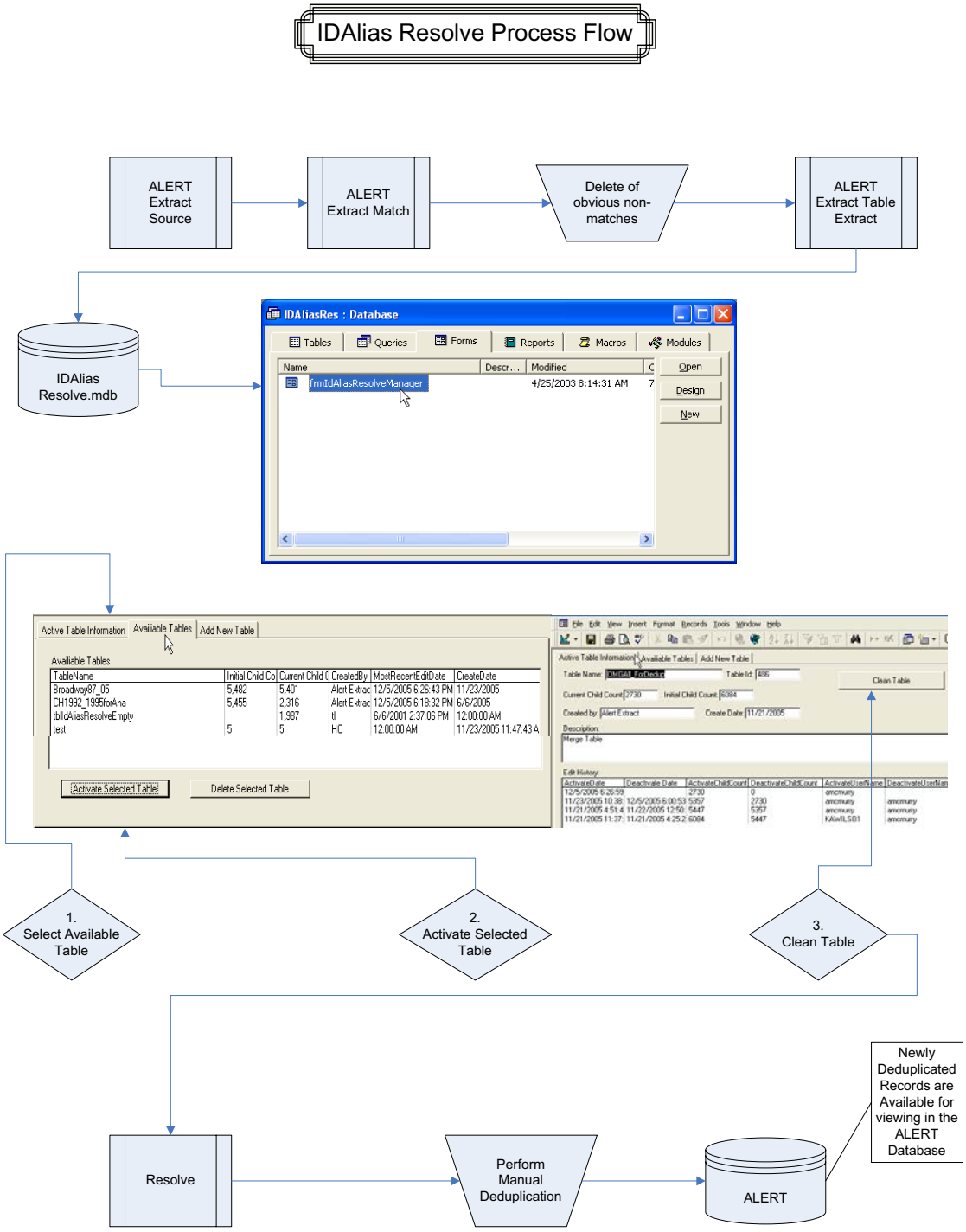
Appendix D: Alert Data Input/Output and Deduplication Flow Chart[43]



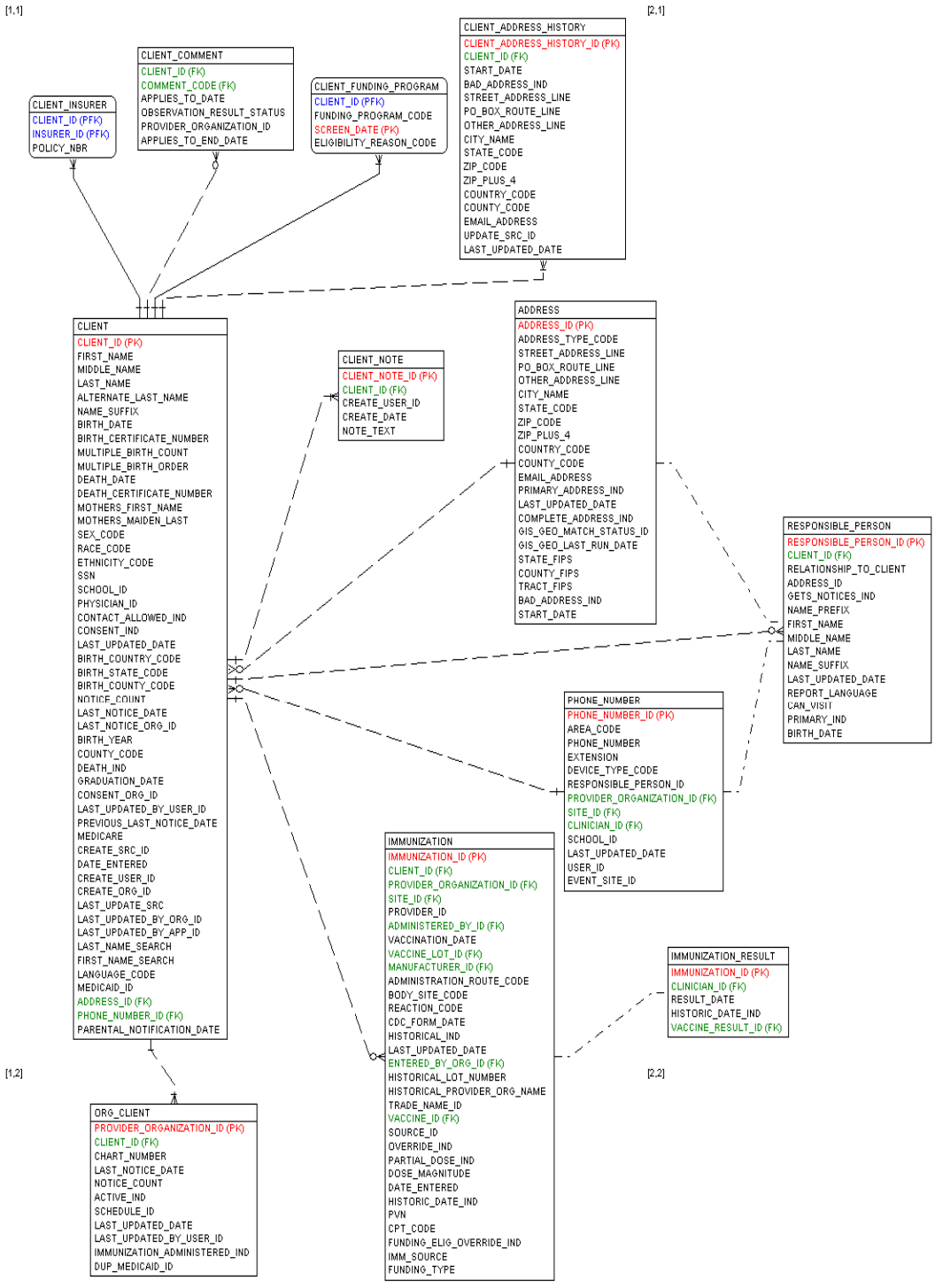
Appendix E: Import/Export & Resolve Process Flow Chart[43]



Appendix F: Resolve Process Flow Chart[43]



Appendix G: New Alert-IIS Data Dictionary



Appendix H: Example Run/Match Deduplication

EX 1:
Incoming Client Info:
==> RIKESHA.S.REIDER b.06/01/1991 f to . 92 WINETREE HILL, ITHACA, NY 14850
RIKESHA: considered 2 links...
REIDER: considered 138 links...
birth 06/01/1991: considered 83 links...

Candidates evaluated:
candidate 0: client_id 1807, ind 21, be 14, fam 25, comp 60
? 1807 RIKESHA..REIDER, born 06/01/1991? to MILLERSON, 92 WINETREE HILL, ITHACA, NY 14850
f=21b m=0/0 l=16b dob=..14 ge=0 a=15 m1=0 m2=0a md=..0 t=0 e=0/0 dod=0 id(:0/0/0)
sum score: 60

candidate 2: client_id 1806, ind 21, be 14, fam -10, comp 25
? 1806 RIKESHA..GROVER, born 06/01/1991? to , 44 CARDINAL, HEUVELTON, NY 13654
f=21b m=0/0 l=-8g dob=..14 ge=0 a=-2 m1=0 m2=0a md=..0 t=0 e=0/0 dod=0 id(:0/0/0)
--danger zone candidate (25)--

candidate 3: client_id 1809, ind -4, be 7, fam 8, comp 11
? 1809 REBA..REIDER, born 06/02/1991f to PAULA ZURICH, 920 LAKE, HEUVELTON, NY 13654
f=-5i m=0/0 l=10b dob=mu7 ge=1 a=-2 m1=0 m2=0 md=..0 t=0 e=0/0 dod=0 id(:0/0/0)

Match found:
matched clients: 1 1807

EX 2:
Incoming Client Info:
==> CALEB.STEPHEN.RUTH-TWEENEY b.02/06/2005 m to KRISTIN.TWEENEY 11 BIRCH ST, MOREAU, NY 12803
CALEB: considered 2114 links...
RUTHTWEENEY: considered 5 links...
birth 02/06/2005: considered 103 links...

Candidates evaluated:
candidate 0: client_id 3423, ind 15, be 14, fam 16, comp 45
? 3423 CALEB.S.TWEENEY, born 02/06/2005m to TWEENEY, 11 BIRCH STREET, FORT EDWARD, NY 12828

f=10b m=5/0 l=10e dob=..14 ge=0 a=-2 m1=0 m2=8b md=..0 t=0 e=0/0 dod=0 id(:0/0/0)
sum score: 45

candidate 2: client_id 3425, ind 10, be 14, fam 15, comp 39
? 3425 CALEB..RUTHTWEENEY, born 02/06/2005m to , , SCHROON LAKE, NY 12870
f=10b m=0/0 l=17c dob=..14 ge=0 a=-2 m1=0 m2=0a md=..0 t=0 e=0/0 dod=0 id(:0/0/0)
sum score: 39

candidate 5: client_id 6754, ind 10, be 14, fam -10, comp 14
? 6754 CALEB..SOCKLEY, born 02/06/2005m to , 70 ELM, AKRON, NY 14001
f=10b m=0/0 l=-8g dob=..14 ge=0 a=-2 m1=0 m2=0a md=..0 t=0 e=0/0 dod=0 id(:0/0/0)

Match found:
matched clients: 2 3423 3425

Appendix I: CDC Evaluation Data Set Information[42]

Data Input Format

Item Number	Field	Min and Max Length (number of characters as occurs in the actual test data)
1	Patient Last Name	Min:3 Max:15
2	Patient First Name	Min:1 Max:14
3	Patient Middle Name	Min:0 Max:11
4	Patient Suffix Name	Min:0 Max:3
5	Patient Date of Birth (yyyymmdd)	8
6	Patient Gender ("M", "F", or "U")	F
7	Mother Maiden Name	Min:0 Max:10
8	Mother Last Name	Min:3 Max:15
9	Mother First Name	Min:2 Max:9
10	Mother Middle Name	Min:0 Max:9
11	Vaccine Name	Min:3 Max:9
12	Vaccine Code (CVX code)	2
13	Vaccination Date (yyyymmdd)	8
14	Vaccine Manufacturer (MVX code)	3

Data Output Format

Field Number	Field Description	Valid Values
1	Unique Patient (Client) ID Assigned by Registry	Number between 0 and 999999999
2	Questionable match flag ("Y" indicates this record may be a duplicate and requires human review).	"Y" or "N" (Leave as "N" if flag not used by the registry)
3	Patient Last Name	Text (max length 50)
4	Patient First Name	Text (max length 50)
5	Patient Middle Name	Text (max length 50)
6	Patient Suffix Name	Text (max length 15)
7	Patient Date of Birth	YYYYMMDD
8	Patient Gender	"M" or "F" or "U"
9	Mother Maiden Name	Text (max length 50)
10	Mother Last Name	Text (max length 50)
11	Mother First Name	Text (max length 50)
12	Mother Middle Name	Text (max length 50)
13	Vaccine Code	Number (2, 3, or 4 digits) (CVX code)
14	Vaccination Date	Yyyymmdd
15	Vaccine Manufacturer	Text (3) (MVX code)

Duplicate Problem Types

Duplicate Problem Types	Description
First Name Spelling	Nicknames, typos, or variations of first name. These can sometimes match by soundex or partial matching.
Last Name Spelling	Typos or misspellings of last name. These can sometimes match by soundex or partial matching.
First Name Hyphenation	Hyphenated first name has missing hyphen or missing one part of name.
Last Name Hyphenation	Hyphenated last name has missing hyphen or missing one part of name.
First Name Reversed w/Last Name	First name has been reversed with last name; for some names not easy to distinguish
First Name Reversed w/Middle Name	First name has been reversed with middle name
Middle Name Reversed w/Last Name	Middle name has been reversed with last name.
First Name as “Baby”, “Baby Boy”,…	Child has been entered into the system possibly with hospital data prior to naming.
Date of Birth Difference	Date of birth for same person does not match due to error in day, month, year or some combination of these.
Gender Difference	Gender for same child does not match other record due to error.
Different Last Name	Last name is totally different due to re-marriage, foster care, or other reasons.
Suffix included in First Name	Suffix erroneously included in first name field
Suffix included in Last Name	Suffix erroneously included in last name field
Duplicate Core Data (first,last,DOB,sex)	The first name, last name, date of birth and gender fields are identical in both records although other fields may not completely match. These cases are common and normally not considered a problem by registries.
Exact Duplicate (all demographic fields)	Every demographic field is an identical duplicate. (This includes the child names, mother names, DOB, & gender. Some may even have identical vaccines as would occur when an electronic submission is re-sent.) These cases are common and normally not considered a problem by registries.

Catagories of Non-duplicate Test Data That Appear to be Duplicates

Non-duplicate Type	Description	Notes
First Name Spelling	For each case, two records have same last names and same DOB but they have first names spelled different and mothers are different.	Records could be confused as same person with spelling first name problem.

Last Name Spelling	For each case, two records have same first names and same DOB but they have last names spelled different and mothers are different.	Records could be confused as same person with spelling last name problem
First Name Hyphenation	For each case, two records have the same DOB and last name and have a first name hyphenation difference. Mothers are different.	Records could be confused as same person with a first name hyphenation problem.
Last Name Hyphenation	For each case, two records have the same DOB and first name and have a last name hyphenation difference. Mothers are different.	Records could be confused as same person with a last name hyphenation problem.
First Name Reversed w/Last Name	For each case, two records have the same DOB and have reversed first and last names from each other. Mothers are different.	Records could be confused as same person first and last name switched.
First Name Reversed w/Middle Name	For each case, two records have the same DOB and last name and have reversed first and middle names from each other. Mothers are different.	Records could be confused as same person first and middle name switched.
Middle Name Reversed w/Last Name	For each case, two records have the same DOB and first name and have reversed middle and last names from each other. Mothers are different.	Records could be confused as same person middle and last name switched.
First Name as “Baby”, “Baby Boy”,…	For each case, two records have “baby” as first name, same last name, and same date of birth. Other fields differ.	Compares to duplicate cases where “baby” was part of first name.
Date of Birth Difference	For each case, two records are two people with same or similar names and different date of birth and mother data.	Records could be confused as same person with date of birth error.
Gender Difference	For each case, two records have similar first names, same last name, and same DOB but they have different gender and mother data.	Records could be confused as same person with an error in gender code
Different last name	For each case, two records have same first names and same DOB, but they have different last names and mother data.	Records could be confused as same person with different last name
Duplicate Core Data (first, last, dob, sex)	For each case, two records have same first and last names, date of birth, and gender - but different middle names and mother data.	Records could be confused as same person.
Siblings	For each case, two records represent two brothers and/or sisters. All fields match except DOB, first and middle names. (Mothers are the same).	Records could be confused as same person with first name and DOB errors.
Multi-births	For each case, two or more records represent twins and triplets. All fields match except for first name, maybe middle. (Mothers are the same).	Records could be confused as same person with first name error.

First Name Spelling AND different DOB	For each case, two records have same last name but first name and DOB have some differences. (May still be similar but not exact match). Mothers are different.	Records could be confused as same person with first name and DOB errors.
Reverse first and middle AND different DOB	For each case, two records have different DOBs and reversed first and middle names. (May still be similar but not exact match). Mothers are different.	Records could be confused as duplicate with reverse first/middle and DOB errors.
Soundex match	For each case, two records have same DOB. First and/or last names will match based on Soundex but don't really look that much alike. (E.g. Morgan and Morrison). Mothers are different.	Records could be confused as same person with first name and/or last name errors, if too much reliance on Soundex.
Cousins	For each case two records have similarities in some fields – could be last names or mom's names	Records could be confused as the same person with first name, last name, and DOB errors.