An Overview of Architectures and Techniques for Integrated Data Systems (IDS) Implementation

Prashant Kumar

Managing Principal
Integrating Factors Inc.

Contents

Introduction

Health and Human Service (HHS) enterprises typically have multiple administrative data systems that are either custom built or acquired from third parties and then customized. It is not uncommon for HHS enterprises in larger jurisdictions to implement ten or more administrative data systems for managing diverse programmatic needs such as public welfare programs, homeless services, mental health services, and drug and alcohol abuse services.

There are a number of reasons for this. First, in the health and human services world, data systems are typically built around programmatic requirements and funded through program-oriented funding streams. As new programs are developed and expanded, program funds are utilized to develop or modernize the supporting information systems. Therefore, to some extent, the application portfolio of the HHS enterprise reflects the evolution and development of its programs.

Second, it would be difficult to build a single large data system that can incorporate the myriad administrative requirements of various programs. The functional and regulatory requirements of multiple programs would be challenging for a single system to support. Once built, these monolithic data systems would be even harder to maintain in the face of changing regulatory requirements and agency policies and practices.

Thus, the preferred and practical approach has been to develop data systems that address the needs of a single program or a set of related programs.

There are, however, important consequences of this approach, notably:

- While the program-centric emphasis of the data systems allows the operational, fiscal, and regulatory requirements to be well supported, it can cause the data systems to fall short in supporting the informational needs of the caseworkers and providers. Caseworkers and service providers typically have a problem-solving focus that requires them to comprehensively identify, assess, and mitigate risks. The information provided by the programmatic data systems is generally agency- or program-specific and doesn't provide a comprehensive view to aid the caseworkers in assessing risk, identifying care gaps, and planning and delivering services in coordination with other agencies.

- The program-centric emphasis of the data systems makes it more difficult for policy makers and senior management to gain insights for policy-level decision-making. The lack of an end-to-end view makes it harder to understand the chain of consequences that leads to specific positive or negative outcomes, making it harder to identify and assess policy options.

Clearly, HHS enterprises must achieve a balance between the operational and administrative requirements of the program and the informational requirements of the problem-solving audience. An integrated data platform that collects administrative data from program-centric data systems, links them together by matching data about individuals and families across service systems, and makes an integrated data view available for use by the problem-solving audience is a viable approach to achieving that balance.

This paper begins with a discussion of potential applications of linked administrative data in policy-level and in case-level decision-making. It then presents the primary data integration approaches and options that are available to the HHS enterprises based on today's technologies and know-how. The paper also addresses the data architecture options and business process implications of embarking on a data integration program.

Applications of IDS Data

The following are the typical applications IDS population-level data:

- **Policy Analysis:** Gaining insights into client populations and service patterns to identify new policy initiatives.
- **Target Population Identification and Stratification:** Using cross-agency service and outcome patterns to identify new targeting approaches, refine intervention strategies, and develop differential treatment protocols.
- **Utilization Analysis:** Analyzing cross-agency service utilization patterns to optimally allocate scarce programmatic and fiscal resources.

The following are some examples of how IDS client-level data can be used by human services professionals:

- **Assessment:** Using cross-agency data on past assessments, interventions, and their outcomes to develop deeper insights into the dynamics of a case; identifying the events and risk factors that have contributed to current outcomes and assessing the effectiveness of potential interventions in the future.
- **Service Coordination:** Using cross-agency data about current and planned services to identify care gaps and overlaps; coordinating the planning and delivery of services with peer agencies to minimize the care gaps and overlaps; sequencing the delivery of services to maximize the odds of positive outcomes.
- **Cross-Agency Alerts:** Identifying client's touch points with other agencies to notify them of significant events: For instance, public welfare system notifying child protective services of a change in the family composition of a currently active case; corrections system sending release notification to the public welfare system for client's Medicaid eligibility to be re-established, etc.

Data Integration Approaches

Health and Human Service (HHS) enterprises have long recognized the need to share data across program and agency boundaries. Data integration efforts ranging from basic need-based data matching to fully-automated real-time data sharing have been undertaken through the years. In this section, we will examine the three primary integration approaches in practice today.

- Need Based Data Integration
- Periodic Data Integration
- Continuous Data Integration

Most real-world implementations do not fit neatly in one of the above categories but rather are hybrids of more than one. Nevertheless, understanding the attributes of each approach is essential to understanding the available solution options.

NEED BASED DATA INTEGRATION

Today, because of the systems separation, most cross-agency or cross-program data analysis efforts require manual sharing of administrative data. Analysts begin by identifying the administrative datasets that can be utilized to address a particular business or analytic need at hand. Once the required datasets have been identified, and the necessary programmatic and legal approvals obtained, data is collected, organized, and prepared for analysis.

The key element of this approach is that the data matching effort is undertaken to address a specific business or analytic need, and the data preparation tasks such as data profiling, cleansing, transformation, matching, and linking are performed in the context of the particular business problem or analytic need. For example, a child welfare agency wanting to better assess the needs of its aging out population may be interested in information about living arrangements of its past clients. For this particular analytic need, the agency may want to match its client datasets with those of the homeless system.

Advantages

- The approach is effective in instilling a culture of cross-agency data sharing and coordination.
- It helps establish the governance structure and processes prior to embarking on more ambitious programs.
- It helps ascertain the data availability and quality levels at peer organizations.
- Benefits can accrue much more quickly as compared to the other approaches requiring complex technological implementations.

Disadvantages

- Each data analysis effort is treated as a one-off project requiring its own legal review and approval. Depending upon the type of data being shared, the programmatic and legal review process can take anywhere from a few days to several months.
- Although the experience gained by the data analysis staff speeds up the data preparation process over time, the lack of a repeatable and automated process requires a considerable amount of time to be spent on preparing the data.

- Because the manual data matching efforts are treated as one-off tasks, necessary processes and controls to ensure confidentiality of data are often inadequate.
- This approach can't be used as a reliable data sharing mechanism for case-level work. Without a data system to broker the cross-agency data exchange, the processes for client confidentiality related legal requirements would be prohibitively complex and expensive to implement.

PERIODIC DATA INTEGRATION

With the Periodic Data Integration approach, HHS enterprises implement a process of collecting, cleansing, organizing, and storing data at pre-determined frequencies such as monthly, quarterly, or annually. Unlike the Need Based Data Integration approach in which cross-agency data matching is carried out after a particular analytic need has been identified, Periodic Data Integration seeks to pre-integrate data for a class of business problems or analytic needs.

With this approach, participating organizations develop data sharing agreements to formalize their data sharing relationships. A data sharing agreement is a formal contract that identifies specific data categories to be shared, the obligations of the participating organizations regarding the purpose for which data can be used, the frequency with which data will be made available, client confidentiality requirements, data security requirements, etc.

Advantages

- The availability of pre-integrated data greatly reduces the data preparation time and decreases the cycle time for analytical work. Specifically, programmatic and legal reviews can be expedited and often are not necessary if the data sharing agreement is carefully developed to include broad classes of analysis for which the data can be used.
- Unlike the need-based data sharing approach, the periodic approach emphasizes streamlining of processes and system automation, allowing the data sharing program to mature with improved processes for data quality management, confidentiality management, service level management, etc.

- In contrast to the Continuous Data Integration approach described later in this section, this approach allows a window for data reconciliation and reasonability testing before the data is released for analysis.

Disadvantages

- Due to the inherent latency of the periodic data integration process, this approach may not be appropriate as a data sharing solution for case-level work that typically requires near real-time data.
- This approach must address the complexities of implementing broad data security measures for its data repositories including access roles, data usage, regulatory requirements, etc.

CONTINUOUS DATA INTEGRATION

With the Continuous Data Integration approach, participating organizations develop a shared information source that is continuously kept current with the administrative systems. Unlike Periodic Data Integration, in which large datasets are brought together at regular intervals, Continuous Data Integration involves immediate posting and constant mirroring of administrative data in the shared data repositories. The key advantage of the Continuous Integration approach is that it makes integrated data available not only for research and policy purposes but also for casework and operational planning purposes.

Continuous Data Integration must also address certain complexities that result from the real-time processes. In particular, unlike Periodic Integration, there is no batch window available to complete data quality checks, source to target data reconciliations, and other quality control functions. Thus, the data quality checks and measurements must be implemented as pre-defined business rules.

Further, because data is brought into the shared information repositories while it may still be volatile, the IDS must recognize and process any changes when they occur. For example, if an incorrect social security number is entered for a client at a source agency and is then corrected a few hours later, the associated impact on data matching or linking must be determined and corrected through an automated process. Such complexities do not exist in Periodic Integration that primarily deals with non-volatile data.

Advantages

- Data is current and can be used for case-level work.
- Like the periodic approach, Continuous Data Integration allows data to be pre-integrated, shortening the data analysis time. Shorter cycle time for data analysis allows the analytical process to be iterative.

Disadvantages

- The additional functional complexity and data management requirements lead to higher systems development and maintenance costs and longer system implementation time.
- If the scope of the IDS includes support for client-level work, there will be additional complexities of business processes for client consents, waivers, court orders, and the necessary administrative and legal reviews.
- A single integrated database increases the impact of any security breach that may occur.

The "How-To" of Integrated Data Systems

IDS deployments require a number of design and implementation issues to be addressed, such as where integrated data would be stored, how data would be moved from the administrative data sources to the destination database, what methods would be used to match and link data about clients and service providers across data systems, what technologies would be used to deliver data to the decision makers, etc. In this section we will discuss the methods, design patterns, options, and techniques available to successfully implement an IDS. Specifically, we will address:

- Data Architecture
- Data Matching and Linking
- Data Retrieval/Delivery
- Data Security
- Technologies and Tools
- Data Governance Process

DATA ARCHITECTURE

Depending upon factors such as intended use of the IDS (policy and/or case-level work), the complexities of legal requirements associated with data being shared, the type of data being shared (client identifying data, service history data, unstructured data such as case notes and documents), a number of data architecture choices are available including:

- Data Warehouse
- Federated Data
- Hybrid Approach

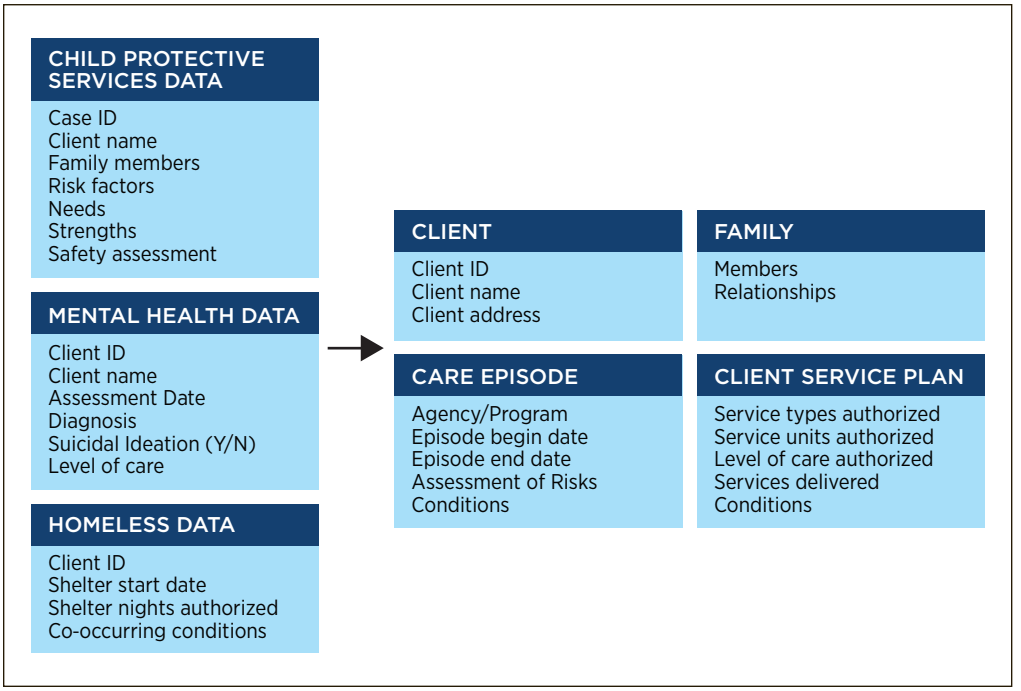
Data Warehouse

Transactional systems such as the administrative data systems in HHS enterprises are designed to maintain integrity of the database transactions and avoid "update anomalies" caused by duplication of data. In the design process of a transactional system, the logical data modeling process step ensures that data structures are "normalized." In other words, duplication of data is avoided by storing each data element in only one place. Data normalization is achieved by storing "keys" to fragments of data that are stored in one place and referenced from related fragments. By avoiding the update anomalies, data normalization allows the transactional systems, such as the HHS administrative data systems, to maintain data and transaction integrity. The design of the transactional systems doesn't directly lend itself to delivering holistic client information because data is fragmented across multiple systems, databases, and data structures.

Data warehouses can be used as the means to bring client data together to directly support the needs of decision makers. A data warehouse collects data from multiple administrative systems, links them together, and stores them in a centralized repository. The centralized repository utilizes specialized data structures that are optimized to directly support the informational needs of the decision-makers. The specialized data structures typically include a set of "Star Schema" that utilize highly de-normalized dimensional data' to quickly answer complex queries of the business decision-makers.

A number of IDS implementations have successfully used data warehouses for policy analysis and research.

FIGURE 1: IDS Data Warehouse - From Program Context to Client Context



As shown in Figure 1, while the distinct administrative data systems store client data in a functional or administrative context, the IDS data warehouse integrates client data across functional areas into entities such as Client, Family, Care Episode, and Service Plan. In addition, the IDS data warehouse would retain historical data and maintain consistent cross-agency reference data about service provider organizations, types and names of services, and types of outcomes. These characteristics of the data warehouse, combined with its reporting and analysis capabilities, make it an attractive design option to support policy analysis and research.

The design of IDS data warehouses requires judicious trade-offs. IDS data warehouses must be designed to support broad classes of analytical work, not just a specific analytic need. This presents a design challenge–too broad a scope for the data warehouse would require large amounts of programmatic data to be brought in, resulting in a large and difficult-to-maintain database. On the other hand, too narrow a scope can be constraining for policy analysis and research work. Achieving the right balance between the two extremes is a critical success factor for IDS data warehouse implementations.

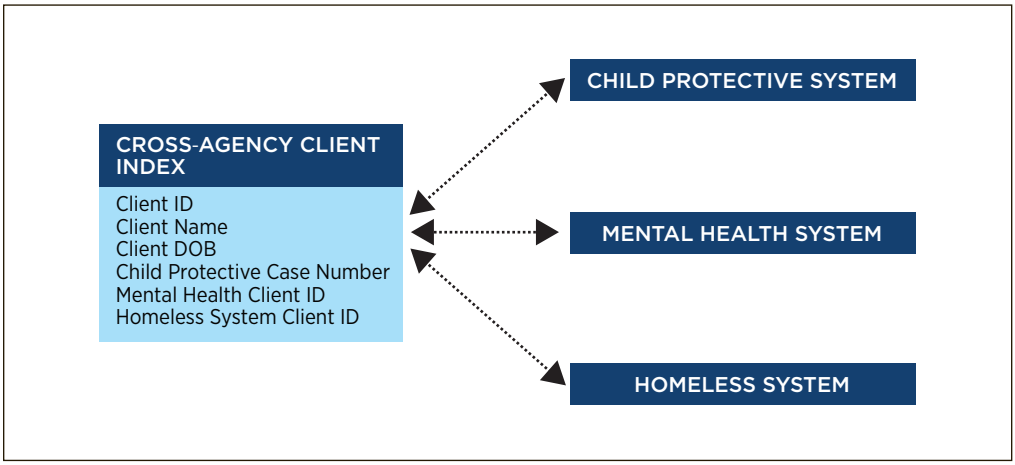
Federated Data

Federated data architecture allows data stored in the transactional data systems, such as HHS administrative data systems, to be dynamically extracted, linked, and presented to the user. This approach obviates the need to store all of the data in a shared database. Instead, federated data systems use specialized software to "expose" data in the transactional data systems to the authorized users or computer programs directly. When a user submits a data request, the federated data system decomposes the user's query into a set of queries and dispatches them to the transactional data systems. The data returned by the transactional data systems is linked and delivered to the requesting user or computer program.

Federated systems maintain one or more cross-agency indexes in order to accurately decompose a user request into a set of transaction system queries. For example, a cross-agency client index might include identifying attributes such as name, SSN, and date of birth, along with the system identifiers for each administrative system that has information about the client.

In figure 2 below, the Cross-Agency Client Index is being used to identify the administrative data systems and the primary keys for the client data within those systems.

FIGURE 2: Using Cross-Agency Client Index to Decompose a User Query in a Federated Data System



In the above example, once the administrative data systems and the associated client locators have been identified, queries are constructed for individual administrative data systems and dispatched to them. The mechanism used for dispatching the query may be one of the following:

- For administrative data systems that have been built using service-oriented architecture (SOA), a published web services-based interface may be available. SOA interfaces are based on open standards and support the exchange of data contained in XML documents over a commonly available protocol such as HTTP.
- For legacy data systems that do not have a published interface, one may need to be built using the Enterprise Application Integration (EAI) approach. In contrast to the SOA approach, which enables interoperability and data exchange using fine-grained web services, the EAI approach utilizes "adapters" that wrap legacy system functionality as software interfaces. The IDS software can use the EAI interface to acquire data from the legacy systems and make it available to calling applications.
- For legacy data systems that can't support EAI adapters, other approaches, such as terminal emulation, may be viable options although they are not as reliable as other options.

There are a number of factors that make the federated data approach an attractive option for making integrated data available for case-level decision-making purposes, notably:

- Because the majority of the client data in the federated approach stays in the programmatic data systems, the legal requirements related to management and disclosure of sensitive client data can be more directly enforced. For example, when a child welfare case results in adoption, requiring that the pre- and post-adoption data about the child be separated by all data systems, the federated approach would automatically meet the requirement without any specialized processing. In other words, a query sent to the child welfare system through the federated data interface would not return any pre-adoption information. In contrast, supporting this legal requirement with the data warehouse approach would require the enforcement of adoption related business rules within the data warehouse in order to separate the pre- and post-adoption data.

- With the federated approach, client data from the different administrative systems can be delivered to the user in an appropriate programmatic context and vernacular. This is achieved by presenting client data from different agencies side-by-side using agency-specific terminology. This is harder to achieve with the data warehouse approach that stores data using generalized data structures and terminology.

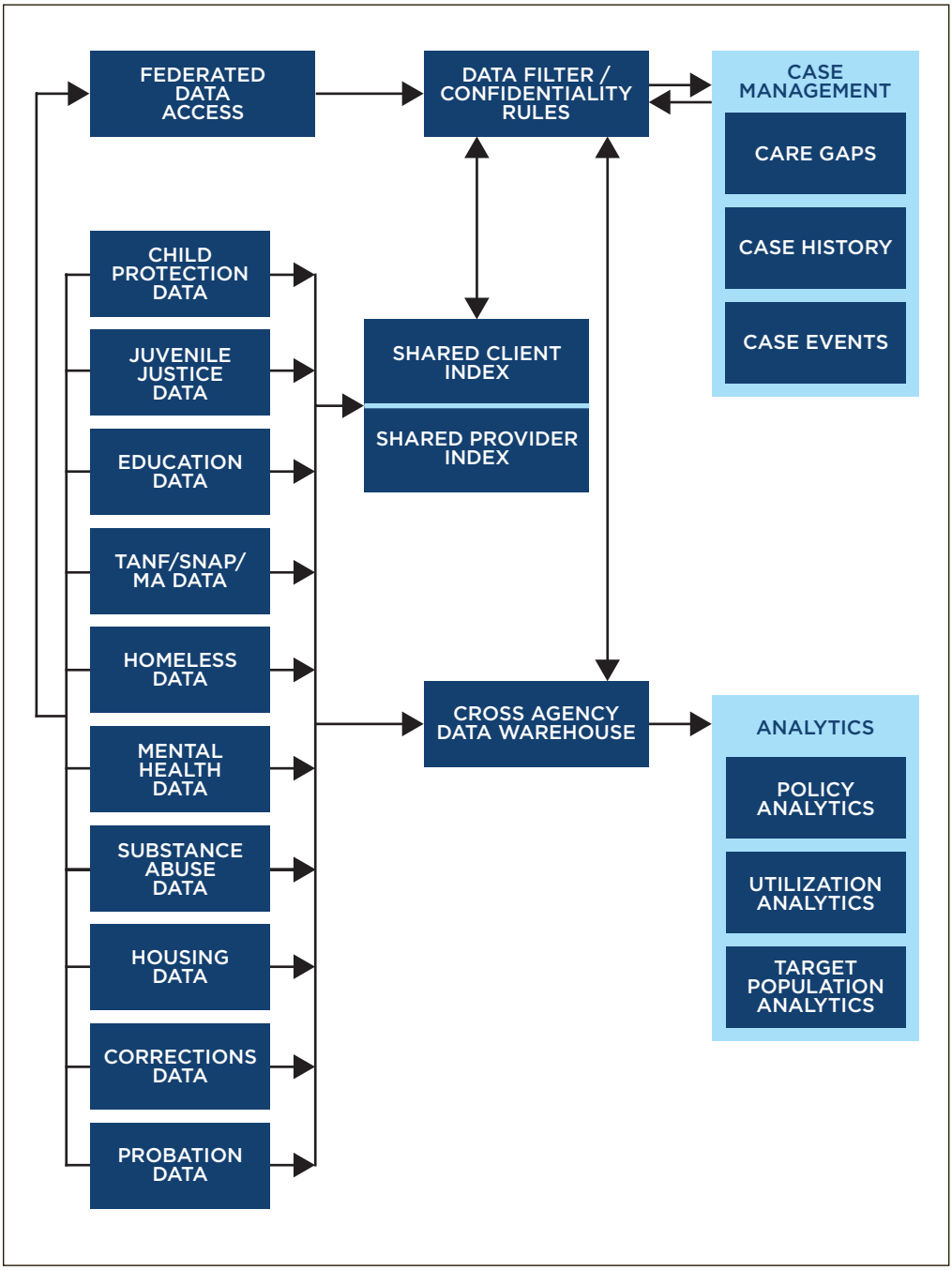
There are also some disadvantages of the federated approach, particularly:

- Because the federated approach relies on obtaining data from the administrative data sources "on demand," any outages in one or more source system would prevent client data from those systems to be presented to the users.
- Policy and research related data analysis tasks require integrated data to be stored in a repository. Federated architecture, therefore, can't be used to support large scale data matching and analysis for policy and research purposes.

Hybrid Architecture

Hybrid architectures that combine the architectural approaches described earlier are also feasible. For example, as shown in Figure 3, a federated architecture is being used to enable cross-agency data exchange for case management purposes while a cross-agency data warehouse is being used to meet the policy and research related needs.

FIGURE 3: A Hybrid Approach—Federated Data Strategy for Case Management and Data Warehouse Integration for Policy and Population Analytics



DATA LINKING CONSIDERATIONS

Much of the benefit of IDS depends upon the effectiveness of matching of client data across disparate administrative data systems. While each data system may have the necessary client data to create a client record internally, it may not have sufficient information about clients to uniquely identify matching client records at other agencies and data systems. Some of the reasons for the lack of sufficient client identifying data include:

- Agencies have different policies about what constitutes sufficient identifying information. A child protection agency might insist upon properly verified social security number, date of birth, current and prior names and addresses while a homeless outreach agency may only able to collect the first and last names of its clients.
- Data is often collected at times of individual and family stress. This impacts the data collection process and ultimately, the quality of client data.
- Systems are often designed with less than sufficient emphasis on usability and worker productivity causing delays and errors in data entry.

Due to the above constraints, the IDS data matching processes have utilized mathematical models to match client records across data systems. These models have been used with much success for over several decades in a wide range of applications including census data matching, highway safety analysis, etc. Using these techniques for IDS typically involves the following considerations:

a. Model type - deterministic or probabilistic

b. Weight Assignment

c. Match thresholds

d. Blocking approaches

e. Link cascades

f. Adhesion factor of linked records

Probabilistic and Deterministic Models

The process of matching client records across programmatic areas relies on resolving identities of individuals based upon identifying attributes such as name, address, date of birth, and social security number. As discussed above, because administrative

data systems often have missing or erroneous data, an individual's identity might not be reliably resolved by requiring that identifying attributes agree in their entirety. Instead, statistical techniques such as deterministic and probabilistic matching employ a variety of ways to compare identifying attributes to determine the "likelihood" of a match. The likelihood is computed as a match score by assigning points, or weights, for agreements and disagreements between the attribute values depending upon the relative power of the identifying attributes in distinguishing client records.

Once a match score has been computed, records above a predefined upper threshold are considered "links" whereas those below a pre-defined lower threshold are considered "non-links". Records with a match score between the lower and upper thresholds are considered "possible links" requiring clerical determination.

While both deterministic and probabilistic approaches follow the general approach described above, the key distinction between the two approaches lies in the methodology chosen to assign weights and linkage thresholds. The deterministic approach sets agreement weights and linkage thresholds outside of and prior to the linking process, possibly drawing upon past experiences and data matching projects. The probabilistic approach, on the other hand the weights and thresholds based entirely on the data sets at hand. Probabilistic models also scale the weights up or down depending upon the relative frequency of an attribute value. For example, an unusual last name is deemed to have greater identity resolution power than a last name that is common and, thus, is assigned a higher weight.

There are advantages and disadvantages of either approach. In most situations, probabilistic matching can provide more accurate matching but it may make it harder to explain why a match occurred or didn't occur. It is also possible to take a hybrid approach that includes elements of both approaches.

Probabilistic Weight Assignment

Fellegi and Sunter built upon the approach initially suggested by Newcombe and provided the mathematical foundation to identify the "linkage rule" (weight assignments and thresholds) for probabilistic matching. The rest of this section involves a more detailed discussion of the equations and ideas that support this kind of matching. Readers who are more interested in a general discussion may wish to skip ahead to the next section entitled "Match Threshold."

The key ideas of this approach are as follows:

- A probabilistic model of record matching includes a set of data element level comparisons, called a "comparison vector", in order to segregate the records into a matched set M and an unmatched set U. Generally speaking, a "comparison vector" can be an arbitrary set of comparisons such that it yields a "Likelihood Ratio (R)" of a match as follows:

$$R = \frac{\text{Probability of comparison holding true in the matched set M}}{\text{Probability of comparison holding true in the unmatched set U}}$$

Or, using formal notation for conditional probability:

$$R = \frac{P\ (comparison \mid M)}{P\ (comparison \mid U)}$$

For example, if last name and Social Security Number (SSN) are used for matching, the likelihood ration can be computed as:

$$R = \frac{P\ (agreement\_SSN\ \&\ LASTNAME \mid M)}{P\ (agreement\_SSN\ \&\ LASTNAME \mid U)}$$

In this example, if the agreement on SSN and agreement on Last Name are considered conditionally independent (which is a reasonable assumption in this case), the likelihood ratio can then be expressed in terms of marginal probabilities:

$$R = \frac{P\ (agreement\_SSN \mid M)}{P\ (agreement\_SSN \mid U)} + \frac{P\ (agree\_LASTNAME \mid M)}{P\ (agree\_LASTNAME \mid U)}$$

The equation can be represented more succinctly by using symbols to represent the conditional probabilities for the matched and unmatched sets. Let *m* and *u* represent the conditional probabilities of agreement for the matched and unmatched sets respectively, the equation then becomes:

$$R = \frac{m\ (SSN)}{u\ (SSN)} + \frac{m\ (LASTNAME)}{u\ (LASTNAME)}$$

To make it easier to estimate parameters, statistical models typically use a logarithmic scale, yielding:

$$\log R = \log \frac{m\ (SSN)}{u\ (SSN)} + \log \frac{m\ (LASTNAME)}{u\ (LASTNAME)}$$

More generally,

$$\log R = \sum \{\log \frac{m}{u} \textit{ for Agreements, } \log \frac{1-m}{1-u} \textit{ for Disagreements}\}$$

To illustrate the computation of the likelihood ratio, let us consider two scenarios:

1. A pair of records agree on both SSN and last name
2. A pair of records agree on social security number but do not agree on last name

Let us assume that we know the *m* and *u* probabilities of SSN to be 0.95 and 0.08 respectively. Similarly, let the *m* and *u* probabilities of LASTNAME be 0.80 and 0.15 respectively.

Scenario 1:

$$\log R = \log \frac{0.95}{0.08} + \log \frac{0.8}{0.15}$$

$$\log R = 3.57 + 2.42 = 5.99$$

Scenario 2:

$$\log R = \log \frac{0.95}{0.08} + \log \frac{1-0.8}{1-0.15}$$

$$\text{Log}R = 3.57 - 2.09 = 1.48$$

The likelihood ratio thus determined can now be compared with the upper and lower thresholds to determine if there is a match.

The estimation of parameters such as the *m* and *u* probabilities for each identifying attribute is a prerequisite to computing the likelihood ratio. Typically, a training sample is used to estimate the *m* and *u* parameters using the Maximum Likelihood Estimate (MLE) model. MLE is a widely used parameter estimation technique and can generally be applied effectively if the conditional independence assumption holds true.

Match Thresholds

Most data linking approaches are designed to maximize positive dispositions (link or non-link as opposed to possible-link) while keeping the misclassification errors, Type I and Type II, within assigned limits. For an IDS, a Type I error, or false positive error, would represent the erroneous linking of data about two different persons whereas a Type II error, or false negative error, would represent the erroneous non-linking of data about a person.

Typically, a data-linking model would have two thresholds or cut-off points – an upper threshold and a lower threshold with the linking rules set up as follows:

- If the comparison score is above the upper threshold, a link would result;
- If the comparison score is below the lower threshold, a non-link would result;
- If the comparison score is between the upper and lower thresholds, a possible link requiring manual review would result.

Ratcheting down the upper threshold would result in more links albeit with a higher Type I error rate. Conversely, ratcheting up the lower threshold would result in more non-links and a higher Type II error rate. Given that both types of misclassification errors have important programmatic and legal implications, IDS implementations must select the thresholds in a manner that maximizes the program benefits and minimizes the risk of misclassification errors. There are two distinct approaches to ascertaining the level of misclassification error rates in data linking:

- Transformed normal mixture models suggested by Belin and Rubin
- Capture-recapture model to ascertain the prevalence of misclassification errors

A related database design strategy is to include provision in the IDS to allow multiple linkage schemes to co-exist. For example, IDS implementations that support client-level decision-making might ratchet up the upper threshold in order to minimize instances of incorrectly combining records of two different individuals. Policy and research oriented IDS may, on the other hand, ratchet down the upper threshold to reveal more linkages and patterns while keeping misclassification errors within a tolerance range. Although there are important data modeling consequences and associated costs of this approach, the author is aware of at least one IDS implementation that has benefitted greatly from this approach.

Blocking Approach

The computational cost (time) of matching each record with every other is prohibitively high with current technologies. For example, an IDS implementation with two data sources of about 10,000 records each would require over 100,000,000 comparisons to yield a few matching records. The traditional solution to this problem has been to partition the records into blocks of records that may have some potential of a match. For example, creating blocks of like sounding last names would bring the number of comparisons down to a manageable number although it would increase the Type II error rate slightly. To minimize the increase in Type II error rate due to blocking, multiple matching passes with different blocking data elements can be carried out.

Today, blocking approaches don't typically involve sorting and partitioning files. Rather, database search predicates are constructed using the blocking data elements.

Link Cascade

IDS implementations typically acquire data from more than two data sources. Some may have as many as 10 or more data sources. This presents the problem of handling a "link cascade" where record A and record C might be linked to each other not because they match each other, but because they each match record B.

Table 1 depicts an example of a link cascade. The example shows person records from the corrections, child welfare and homeless systems. The matching record column shows that records 1 and 3 both match record 2 although they do not match each other.

TABLE 1: Link Cascade

RECORD #	SOURCE	LAST NAME	FIRST NAME	MIDDLE NAME	SSN	BIRTH DATE	MATCHING RECORD	LINKED RECORD
1	Corrections	Webb	Mary	J	515433219	1/02/72	2	2, 3
2	Child Welfare	Jones	Mary		515433219	1/22/72	1, 3	1, 3
3	Homeless	Jones	Mary		515433291	1/22/72	2	1, 2

Link cascades are desirable because they reveal hidden linkages and patterns. However, they require careful consideration during IDS design so as to ensure that the linking process discovers the link cascades and the discovery is not dependent upon the order in which source records are acquired and matched.

Adhesion Factor of Linked Records

A collection of linked records of a client has more information about the client than what is embodied in any single record of the client. As a result, as new data sources are added to the IDS and new linkages between client records are established, a more complete picture of the client starts to emerge. If the IDS implements a linking process that utilizes all known identifying information from the linked records about the client, the probability of finding new links goes up with each new link. This increased "adhesion factor" of linked client records provides an opportunity to find new matches that are often missed by record-to-record matches.

Table 2 depicts an example of a how the adhesion factor of linked client records allows new matches to be found. In this example, record 1 and 2 have been linked based on record-to-record matching. Record 3 however does not match either record 1 or 2 based on record-to-record matching. Record 3 however is linked to both records 1 and 2 because it matches records 1 and 2 taken together.

TABLE 2: Higher Adhesion Factor Over Time

RECORD #	LAST NAME	FIRST NAME	MIDDLE NAME	ADDRESS	SSN	BIRTH DATE	MATCHING RECORD	LINKED RECORD
1	Webb	Mary	J	123 Main St., Any City, USA	515433219		2	2, 3
2	Webb	Mary		123 Main St., Any City, USA		1/22/72	1	1, 3
3	Jones	Mary			515433219	1/22/72	None	1, 2

Increased adhesion factor allows more links to be found through record-to-linked-set matching. However, it has a number of design and operational implications, viz.

- Ensuring that the linking result is not dependent upon the order in which client records are acquired and processed;
- The lack of a theoretical framework to estimate the level of misclassification errors for record-to-linked-set renders the process of tuning the linking algorithm entirely an iterative process;
- Requires increased monitoring of the matching process to prevent "very large" linked sets that can result from data quality issues.

Manual Overrides

There are instances when it may be necessary to allow manual interventions in the match process. For example, in the case of twins, a number of attributes of two distinct individuals might match leading to a link being created between the two records. Such attributes might include last name, date and place of birth, address, and parents' names. In such cases, it may be necessary for a data steward or a caseworker to intervene and mark the records "keep apart". Similarly, there may be situations when certain records should be marked as "keep together". Subsequent passes of the linking process should not override these markers.

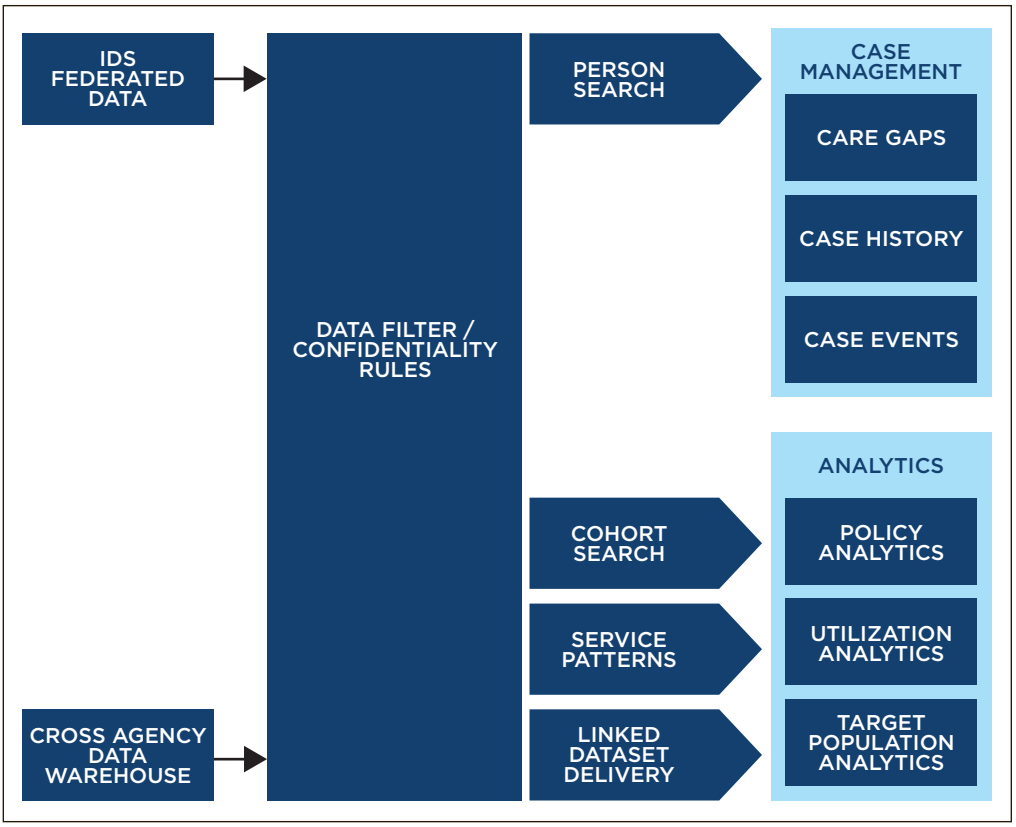
DATA RETRIEVAL

IDS users who are involved in client-level work such as caseworkers, supervisors and contracted providers would access the IDS to get detailed assessment information and service histories about their clients from other agencies and data systems. On the other hand, policy and research oriented users would access client data and service utilization data that may be relevant to their policy or research interests.

Both of these classes of use cases require distinct classes of functionality from the IDS, viz.

- Search
- Data Delivery

FIGURE 4: Data Retrieval from IDS



Search

IDS Search function allows users or applications to initiate data exchange sessions with the IDS. To be effective, the search algorithms must incorporate elements of the data linking process. Further, the search algorithms must also adjust for the variability of data quality levels across programmatic areas.

IDS Search capabilities may include the following:

- **Person Search –** This type of search is usually invoked by applications that support users involved in client-level work such as caseworkers, supervisors and contracted providers. The Person Search function allows a calling user or application to specify search criteria such as name, date of birth, social security number, or address. In response to the search request, IDS Search would conduct a deterministic or probabilistic match to identify client records that match the search criteria.

- **Cohort Search –** Cohort Search allows search criteria for a set of persons to be submitted for cross-agency search. It produces a result set that contains a list of persons whose identifying information matches the search criteria. This type of search is particularly useful for matching client lists between agencies to find common clients. For example, a child welfare agency and mental health agency wanting to collaborate in service planning can identify common clients using Cohort Search.
- **Service Pattern Search –** This is similar to Cohort Search but includes identification and matching of service patterns. For example, a statistician analyzing the cross-agency dynamics of the aging out youth population might want to search for clients with specific patterns of service utilization. Supporting this type of search capability requires a rich user interface that enables users to specify complex service patterns.

Data Delivery

The IDS Data Delivery function allows transactional data such as assessment and service history data to be retrieved about a list of clients. The list of clients may have been developed through the IDS Cohort Search functionality, for example.

Key elements of a data delivery function include:

- **Data Filtering –** Data filtering rules allow sensitive data to filtered out based on factors such as:
    - **Data use case –** Generally speaking, the retrieval of data for client level work is subject to extensive statutory and regulatory requirements related to client privacy and confidentiality. Such information generally can't be released unless an informed client consent, waiver, inter-agency agreement, or court order specifically authorizes its release. Policy and research related usage, on the other hand, has fewer compliance requirements. The Data Filter must take the data use case into account to determine if the usage is for an authorized purpose.

- **Availability of client consent, waiver or court order –** Delivery of sensitive information, particularly information pertaining to mental health history, drug and alcohol services, domestic violence, or HIV, can generally only be delivered for client level work if an informed client consent is on file. Client consents can include provisions for the client or guardian to specify details such as which service systems can share client information, what information categories can be shared, or whether third party providers are allowed to view client information. To apply the authorizations contained in client consents or program level waivers, the Data Filter must implement a metadata-driven mechanism to categorize data elements and associate disclosure rules embodied in the consents and waivers with those data categories. For example, if a consent form has a category to authorize the release of mental health information, the Data Filter must be able to determine what data elements contain mental health information.

- **Linked Data Delivery –** Integrated cross-agency data can be delivered in a variety of structures. Multidimensional star-schema data structures are useful of analytical work using drill-through analysis. Statistical analysis related use cases require longitudinal, flat views of the data over long periods. A well-designed data delivery layer would include options to deliver linked cross-agency data in a variety of structures and formats to support a variety of use cases.

TECHNOLOGIES AND TOOLS

There are a number of technological options, platforms, and tools available to acquire client data from administrative systems. These are described under Data Exchange Technologies and Tools.

After the data has been acquired, it must be cleansed, transformed to a data structure that is suitable for IDS, and integrated with data obtained from other programmatic areas. The technologies and tools for data cleansing, transformation and integration are described under Data Integration Technologies and Tools.

While the Data Exchange and Data Integration technologies help collect and organize integrated data, tools for portal-based presentation, business intelligence, statistical analysis, and data mining help transform data into actionable information to be used by the decision makers. These are described under Information Delivery Technologies and Tools.

Data Exchange Technologies and Tools

Federated architectures typically rely on data exchange at the application layer while the repository-based architectures such as data warehouses exchange data at the database level.

Depending upon the chosen data integration approach (Need Based, Periodic or Continuous), a variety of technologies and tools are available, as described below.

- **File transfer –** File transfer allows a dataset to be sent from one system to another using basic network level services. This is often used for Need Based Data Matching and Periodic Data Integration styles.
- **Message Oriented Middleware (MOM) –** Messaging middleware provides higher-level support (higher than basic network services) to applications in order to exchange information with one another. The support may include guaranteed delivery (messages will be received by receiving application), idempotent delivery (a message will not be delivered more than once), etc. The messaging middleware can be used by both repository based and federated data implementations.
- **Connection Oriented Middleware –** While message oriented middleware enables asynchronous communication between applications, connection oriented middleware enables synchronous communication between applications. Connection oriented middleware is often used for implementing composite applications for data delivery such as a case worker portal.
- **Enterprise Application Integration (EAI) Suites –** EAI suites consist of a data integration server and a set of "adapters" that enable an application to pull data out of another application. There are a variety of adapters available for different database technologies, third party application suites, etc. EAI suites are often used in the federated architecture approach.

- **Web Services –** web services facilitate sharing of data using standards-based, open data exchange. Web services may be implemented over connection-oriented or message-oriented middleware. In order to use web services for IDS integration, data in the administrative data systems may be exposed to calling applications through a published web service interface.

Data Integration Technologies and Tools

These are tools to acquire, analyze, validate, cleanse, detect change, extract, move, replicate, transform, and load data. The primary ones are:

- **Data Profiling -** These tools evaluate a data set, and sometimes multiple related data sets, and provide a summary "profile" of the data content of each column. Statistics such as percentage of missing values, distinct values, distributions, and data types are provided. These tools can provide a quick assessment of the quality of data in the administrative data systems and help identify approaches for data cleansing and standardization.
- **Data Cleansing -** These tools can evaluate data to standardize the format (for addresses, phone numbers, spelling of common last names, etc.) and/or to assign missing components. Some require a knowledge base, such as a list of all valid addresses, or a business directory to cleanse names.
- **Identity Matching –** Identity matching tools allow individual and institutional (provider) identities to be matched using a number of criteria such as name, social security or business tax identification number, birth or incorporation date, etc. Most products support probabilistic as well as deterministic matching. Some products are available as data matching products while others are available as embedded functionality within master data management tools or statistical data analysis tools.
- **ETL (Extract, Transform, Load) –** These tools can extract data from a designated source on a designated schedule, cleanse data to improve data quality using specified rules, transform data to a target schema, and load the data to a target database.

- **Data Replication –** Data replication tools are usually part of a DBMS (Database Management System) and are useful for managing failover.
- **Change Data Capture –** There tools can detect changes in a database, extract the changed data, and send it to another system, such as a data warehouse using a messaging middleware.

Information Delivery Technologies and Tools

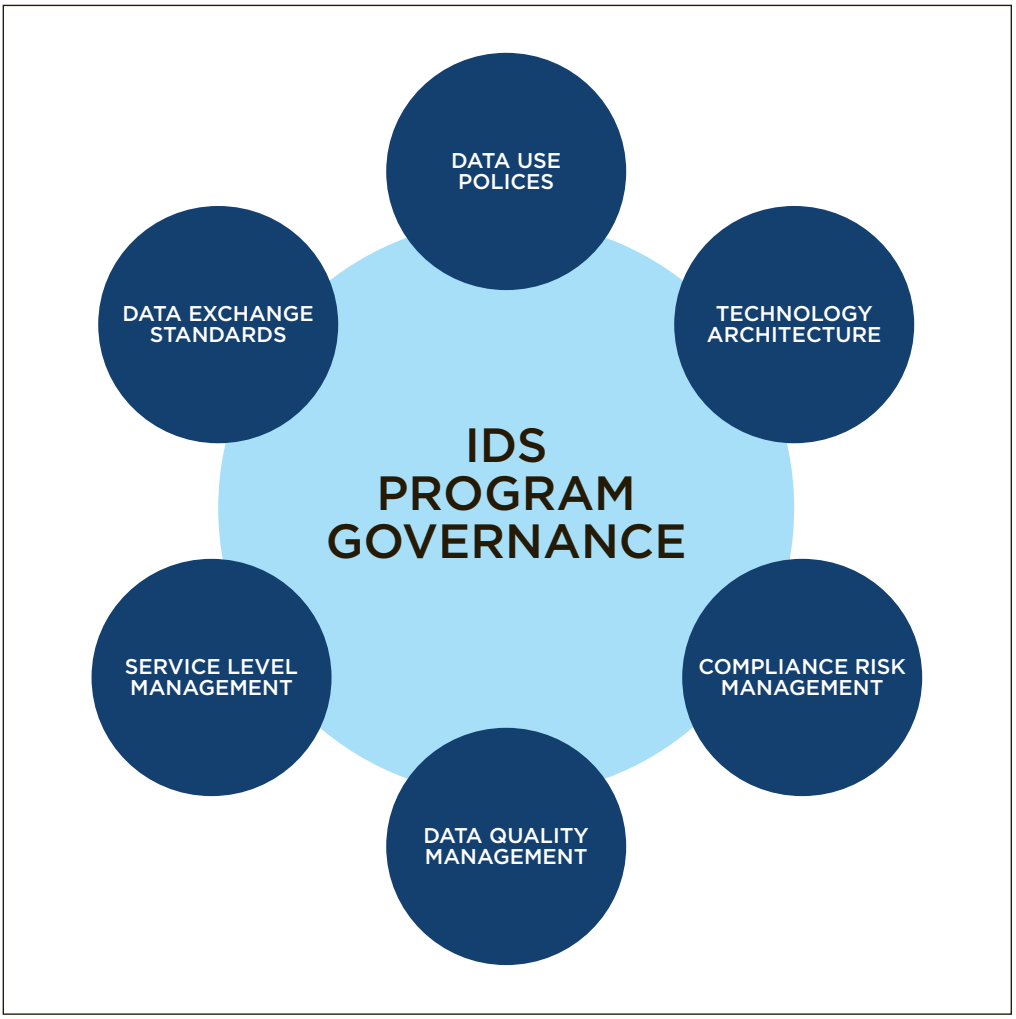
Information delivery technologies and tools allow data in IDS to be transformed into actionable information and presented to decision makers in a context appropriate for their role. This category includes presentation portals, business intelligence tools, statistical data analysis tools, etc., as described below:

- **Portals –** portals allow custom views of data to be easily configured according to user roles and personalized according to user preferences. Portals can be used to deliver information to case level decision makers such as caseworkers and supervisors as well as to policy and research analysts and decision makers.
- **Business Intelligence Tools –** These include reporting and query tools as well as multidimensional data marts and proprietary data cubes for data analysis. A multidimensional design consists of a "fact" table, such as history of client services and associated "dimension" tables such as client, time period, and service type. The multidimensional design enables drill-through analysis functionality for policy operational planning purposes.
- **Statistical Analysis Tools –** These tools support a variety of statistical data analysis approaches to identify correlations and patterns in data. The functionality can include segmentation or cluster analysis, regression analysis, social network analysis, etc.

IDS PROGRAM AND DATA GOVERNANCE

Stakeholders in IDS programs include agency leaders from multiple programmatic areas, human services professionals, policy analysts, data analysts, and providers. With such a diverse stakeholder group, bringing divergent interests into alignment requires executive sponsorship, senior management commitment and formalized processes for conflict resolution and decision-making.

FIGURE 5: IDS Program Governance



An effective governance approach for an IDS program must address the following:

- **Data Use –** The charter for data use policy governance for an IDS program includes data sharing program objectives and, in the context of those objectives, the processes for developing and implementing practices for cross-program data use. Data use policies address the purpose for which specific data categories can be used, data interpretation guidelines, and training requirements for cross-program data use.

- **Compliance Risk –** The charter for Compliance Risk Management identifies the stakeholders and processes for ensuring adherence to privacy and confidentiality requirements. Effective ongoing legal risk management would include processes for monitoring, measuring and controlling the disclosure of sensitive client information.
- **Data Quality –** The charter for IDS data quality management identifies stakeholders and processes for monitoring, measuring and controlling the accuracy, timeliness, consistency and reliability of data. Effective governance of data quality would set expectations, priorities and metrics for data collection and management practices for specific data categories. Cross-program governance of data quality establishes multidisciplinary efforts for data stewardship, data reconciliations, etc.
- **Technology Architecture and Data Exchange Standards –** The IDS architecture essentially includes an ecosystem of cooperating data systems and facilitates the exchange of client data on a timely and consistent basis. The charter for IDS technology architecture would identify stakeholders, processes, roles and expectations for coordinating the data and systems architecture planning the IDS.
- **Service Levels –** As organizations start to rely on data that is generated and managed outside their control boundaries, it becomes important to formalize the expectations, roles and responsibilities of peer organizations. To this end, the IDS program governance would clarify roles and responsibilities of data provider and data consumer agencies.

Cost Considerations

The primary cost factors for an IDS program over the expected life of an IDS would include everything from the acquisition cost during the build and transition phases to the maintenance, operations and training costs on an ongoing basis, as described below.

COST FACTORS

During development IDS programs would incur large one time capital costs including:

- Acquisition cost for software license
- Acquisition cost for hardware and network infrastructure
- IDS software development and implementation costs
- Staff time for business and technical analysis
- Contract administration overheads
- Transition and training costs

In addition to the above one time capital costs, the ongoing operating costs for an IDS would include the following cost factors:

- License maintenance costs – typically 15 to 20% of initial license acquisition costs
- Cost of maintaining and upgrading hardware and network infrastructure
- Software maintenance costs
- Service level management costs
- Staff time for data stewardship, data analysis, and data reconciliations etc.
- Staff time for data governance
- Contract administration costs
- Training costs

Depending upon the size of the jurisdiction and the complexity of its programmatic data systems, the cost of implementing an end-to-end IDS that supports both policy and case level decision-making can vary widely. The variability in IDS implementation costs comes from the following factors:

- Number of data sources
- Complexity of data sources

- Data quality levels of programmatic data systems – data with higher quality requires less transformation during development and less data steward time on an ongoing basis;
- The number and complexity of use cases – casework oriented IDS may have diverse audience but predictable information needs. In contrast, policy and research oriented IDS may have unpredictable and varying information needs requiring a larger operating cost component.
- Legal requirements for data categories to be shared – sharing of sensitive private data in casework-oriented IDS may require support for informed consent processes, additional training and oversight.
- Technology platform – open, standards based platforms generally cost less over time than proprietary platforms.
- Type of IDS architectural approach chosen – for casework oriented IDS, federated architecture often can be quicker and less costly to implement.

Conclusion

Human service organizations are awash with data. But getting the right information in the hands of the right people at the right time has been a challenge. This has resulted in blind spots in the decision-making process at multiple levels. Integrated Data Systems (IDS) can help improve the decision-making process by making quality information about clients available to decision makers at all levels. This can lead to more effective service planning at the client level and more effective policy response at the macro level - with population-level data, policy analysts can develop deeper insights into the cross-agency dynamics of target populations, operations managers can develop a better understanding of the required capacity, resource utilization and costs associated with serving multi-agency clients. With client-level data, caseworkers and service providers can perform more comprehensive assessments and coordinate services with other agencies to minimize gaps in services and maximize the odds of achieving positive outcomes. The technologies and techniques have matured to the point where the cost and risk of IDS deployment are far outweighed by its benefits.

References

Belin, T. R., and Rubin, D. B. (1995), "A Method for Calibrating False- Match Rates in Record Linkage," *Journal of the American Statistical Association*.

Devlin, B.A. and Murphy, P.T. (1988), "An architecture for a business and information systems," IBM Systems Journal. Volume 27, No. 1.

Fellegi, I. P., and Sunter, A. B. (1969), "A Theory for Record Linkage," *Journal of the American Statistical Association.*

Heimbigner, D., and McLeod, D. (1985), "A Federated Architecture for Information Management" *ACM Transactions on Office Information Systems, Vol. 3, No. 3.*

Hohpe, G., and Woolf, B. (2004), "Enterprise Integration Patterns", *Addison-Wesley* 0-321-20068-3.

Inmon, W.H. (1993), "Building the Data Warehouse" *John Wiley & Sons* 0-471-56960-7.

Newcombe, H. B., Kennedy, J. M., Axford, S. J., and James, A. P. (1959), "Automatic Linkage of Vital Records," *Science.*

Project documentation on software and database architecture, DSS CARES (Division of Social Services, Cross-Agency Response for Effective Services), City of Philadelphia (2006).

Project documentation on enterprise architecture, HHS Connect, City of New York, (2010).

Winkler W. Using the EM Algorithm for Weight Computation in the Felligi-Sunter Model of Record Linkage. Washington, DC: Statistical Research Division, Methodology and Standards Directorate, U.S. Bureau of the Census; 2000. Report No.: RR2000/05.