# *An Information Fusion Framework for Data Integration*

Dave McDaniel[1]

*Abstract*

*Despite high demand for and years of dozens of product offerings, enterprise data integration remains a manually intensive effort, with custom development for each data interface. It involves linguistics, ontological models, uncertain reasoning, inference, and other non-exact and not fully understood sciences. This paper presents an approach for making progress in data integration technology by paralleling progress made in the data fusion community where the fundamental problems are now being appreciated. A framework for information fusion as a means to achieve data integration is presented.*

Contents

## 1 Introduction

Information fusion, as used herein, refers to a research and system development community that has been active with conferences and publications for over 15 years. Variously called "sensor data fusion", "sensor fusion", and "data fusion", information fusion deals with paradigms and techniques for "fusing" multi-source data and information. It is defined in DoD as,

> "The synergistic process of associating, correlating, and combining Hostile, Friendly, and Neutral Forces data and environmental factors to derive information and knowledge, tailorable to support the warfighter to effect and expedite command and control." (AC2ISRC, 1999)

Particular techniques and tools deal with optimal estimation (current), smoothing (past), and prediction (future) of information of interest based upon various multiple sources of related information, including measurements, derivations, and references.

Data integration, as used herein, refers to the processes necessary for integrated data warehouses, virtual databases, enterprise databases, knowledge portals, or other forms of multi-input data to be able to be related across the multiple data sources. Translation and transformation techniques and tools are prevalent. It would appear there is an overlap between the areas of concern of information fusion and data integration. The purpose of this paper is to explore the applicability of information fusion paradigms and techniques to data integration.

[1] Author's address is Silver Bullet Solutions, Inc., 4747 Morena Blvd., Suite 350, San Diego, CA 92117, (858) 581-4380, davem@silverbulletinc.com

## 2 Information Fusion

Researchers and system developers in DoD have been working on data fusion problem since the late 1950's. Examples of major systems confronting the data fusion problem were Project Lamplighter and the Simulated Air-Ground Environment (SAGE). In Project Lamplighter, radar measurements of aircraft positions were generated into "tracks" with smoothed position estimates and derived velocity that were then exchanged among three ships where they were correlated into a single air picture. While highly successful, many improvements would be required and developed up to this day, with the Cooperative Engagement Capability (CEC) being the latest incarnation. While conceptually simple, the reality of the data is exceedingly challenging. A major challenge source arises form measurement discrepancies caused by differing radar cross section, radar fade zones, line-of-sight obstructions, multi-path, and differing characteristics of sensors (frequency, pulse type, scan type, signal processing, false-alarm-rate strategy). Other examples of sources for challenges are multi-site and sensor registration errors (navigation, alignment, calibration), target characteristics such as maneuvering, jamming, and deception, and differing fusion processing such as process models, maneuver response, chosen approximations, and sub-optimization strategy. Data fusion problems for other measurement types (e.g., ELINT, SIGINT, IMINT, IRINT, MASINT, HUMINT) and object types (e.g., infrastructure, political) introduced many more challenges. Accurately and precisely estimating the battlespace continues to be one of the most difficult of human endeavors.

One of the landmark advances in the data fusion community was not technical but social. In 1991 the Joint Directors of Laboratories, with input from the community's leaders, developed a data fusion paradigm[2]. This paradigm, shown in Figure 1, provided a framework for communication and coordination amongst the many diverse fusion workers.

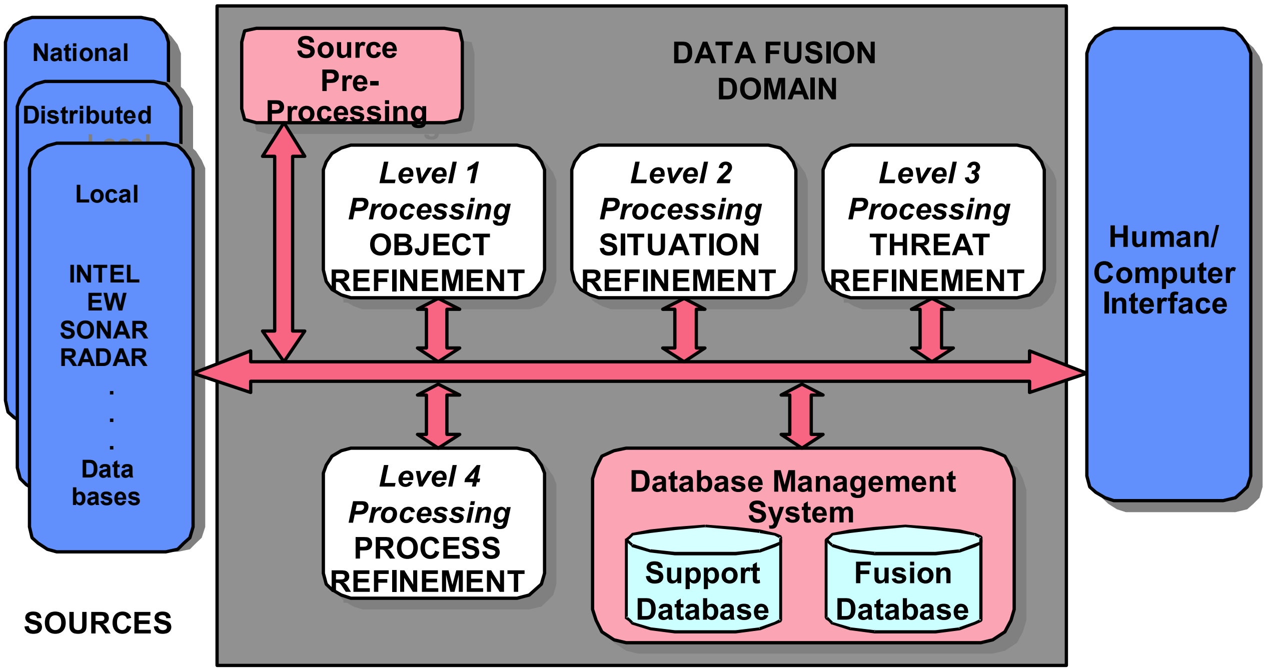

National
Distributed
Local
INTEL
EW
SONAR
RADAR
.
.
.
Data bases
SOURCES
Source Pre-Processing
DATA FUSION DOMAIN
Level 1 Processing OBJECT REFINEMENT
Level 2 Processing SITUATION REFINEMENT
Level 3 Processing THREAT REFINEMENT
Level 4 Processing PROCESS REFINEMENT
Database Management System
Support Database
Fusion Database
Human/ Computer Interface

Figure 1. JDL Fusion Paradigm

[2] *Functional Description of the Data Fusion Process*, Data Fusion Development Strategy, Office of Naval Technology, November, 1991

The levels are defined as follows[3]:

a) Level One Fusion Processing – Object Refinement. Level one processing combines parametric data from multiple sensors to determine the position, kinematics, attributes or identity of low level entities. Key functions include:

   1) Data Alignment – Normalization of data with respect to time, space, and units to permit common data processing.

   2) Data/Object Correlation – Determination of whether newly received observations relates to existing tracks, other contacts, data in the database, or are false data.

   3) Object positional/kinematic/attribute estimation – Combination of data from multiple sensors to determine the value of a state vector (i.e. position, velocity, and attributes) which best fits the observed data. Examples include geolocation and target tracking.

   4) Object Identity Estimation – Determine the classification or identity of entities such as emitters, platforms, or low-level military units, based on attributes or features. Examples are HULTEC and SEI.

b) Level Two Fusion Processing – Situation Refinement. Level two processing develops a description or interpretation of the current relationships among objects and events in the context of the operational environment. The results of this processing is a determination or refinement of the battle/operational situations. Key functions include:

   1) Object Aggregation – Establishment of relationships among objects including temporal relationships, geometrical proximity, communications links, and functional dependence.

   2) Event/Activity Aggregation – Establishment of relationships among diverse entities in time to identify meaningful events or activities.

   3) Contextual Interpretation/Fusion – Analysis of data in the context of the evolving situation including weather, terrain, sea-state or underwater conditions, enemy doctrine, and socio-political considerations.

   4) Multi-perspective Assessment – Analysis of data with respect to three perspectives: (1) the blue (friendly) force, (2) the red (enemy) force, and (3) the white (neutral) – how the environment affects the red and blue.

c) Level Three Fusion Processing – Threat Refinement. Level three processing develops a threat-oriented perspective of the data to estimate enemy capabilities, identify threat opportunities, estimate enemy intent, and determine levels of danger. Key functions include:

   1) Capability Estimation – Estimation of the size, location, and capabilities of enemy forces.

   2) Predict Enemy Intent – Determination of enemy intention based on actions, communications, and enemy doctrine.

   3) Identify Threat Opportunities – Identification of potential opportunities for enemy threat based on prediction of enemy actions, operation readiness analysis, of friendly vulnerabilities, and analysis of environmental conditions.

   4) Multi-Perspective Assessment – Analysis of data from the red, white, and blue perspectives.

[3] Functional Description of the Data Fusion Process", Data Fusion Development Strategy, Office of Naval Technology, November, 1991

5) Offensive/Defensive Analysis – Prediction of the results of hypothesized enemy engagements considering rules of engagement, enemy doctrine, and weapon models.

d) Level Four Fusion Processing – Process Refinement. Level four processing monitors and evaluates the ongoing fusion process to refine the process itself, and guides the acquisition of data to achieve optimal results. These interactions among the data function levels and with external systems or the operator to accomplish their purpose. Key functions include:

1) Evaluations – Evaluation of the performance and effectiveness of the fusion process to establish real time control and long term process improvements.

2) Fusion Control – Identification of changes or adjustments to processing functions within the data fusion domain which may result in improved performance.

3) Source Requirements Processing – Determination of the source specific data requirements (i.e. identifies specific sensors/sensor data, qualified data, or reference data) needed to improve the multi-level fusion products.

4) Mission Management – Recommendations for allocation and direction of resources (sensors, platforms, communications, etc.) to achieve overall mission goals.

5) Source Pre-Processing/Database Management System. Ancillary functions in the context of data fusion processes.

6) Source Pre-processing includes normalizing, formatting, ordering, batching, and compressing input data to satisfy process estimation and processor computational and scheduling requirements. This can also provides special functions such as priority treatment of data with characteristics designated to be of special interest by the user.

7) Data base management systems provide functionality critical to the data fusion process. The fusion database maintains short-term data compiled by the ongoing process regarding objects, situations, and threats. The support database maintains longer-term data relevant to anticipated mission and process demands. This may include reference data, equivalent to that described under sources, but which is known to be relevant to a mission and is, therefore, pre-stored for immediate availability. The support database may also be updated or modified by the fusion database for local usage, whereas modification of source reference data is generally difficult.

Examples of fusion functions for levels 1-3 are shown in Figure 2.

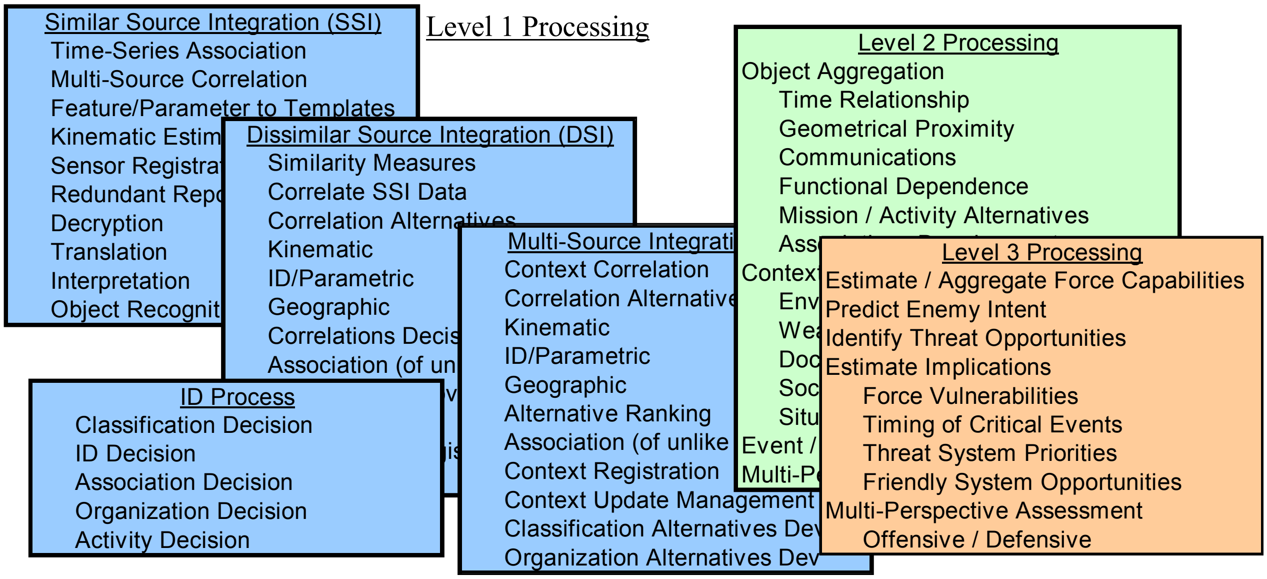


**Figure 2. Example Fusion Functions**

Figure 3 shows a notional level 1 processing flow. On the left is a real object with features (e.g., visible, infrared) and components. The components (e.g., radios, and radars) have observable features as well. The features have observables such as visible light, infrared, radar reflections, and own-radar waves. Multiple sensors can detect and measure the observables. As depicted in the top bank of sensors, some sensors exchange information directly such as in the Navy Cooperative Engagement Capability. Similar Source Integrators (SSI) operate on like-phenomenology or feature observables such an ESM/ELINT, COMINT externals, IR image, IR signature. The rationale for this architecture is that like-phenomenology or feature observables can be directly compared to determine if measurements from multiple sensors are from the same object or component. Also, the SSI's specialize in a specific phenomenology and feature or component model. Some SSI's operate cooperatively over SSI-specific busses such as the TADIL-J ESM subnet. The SSI's produce estimates and hypotheses regarding the components and from them the main object. In some cases of features of the object, rather than components, are operated on by the SSI resulting in hypotheses of the object directly, without reference to components (e.g., EO imagery). The SSI estimates and hypotheses are provided to Dissimilar Source Integrators (DSI) that use object templates to correlate across SSI's. The DSI's can also operate over a bus such as the TADIL Surveillance net. The result is estimates and hypotheses regarding the actual object of interest.

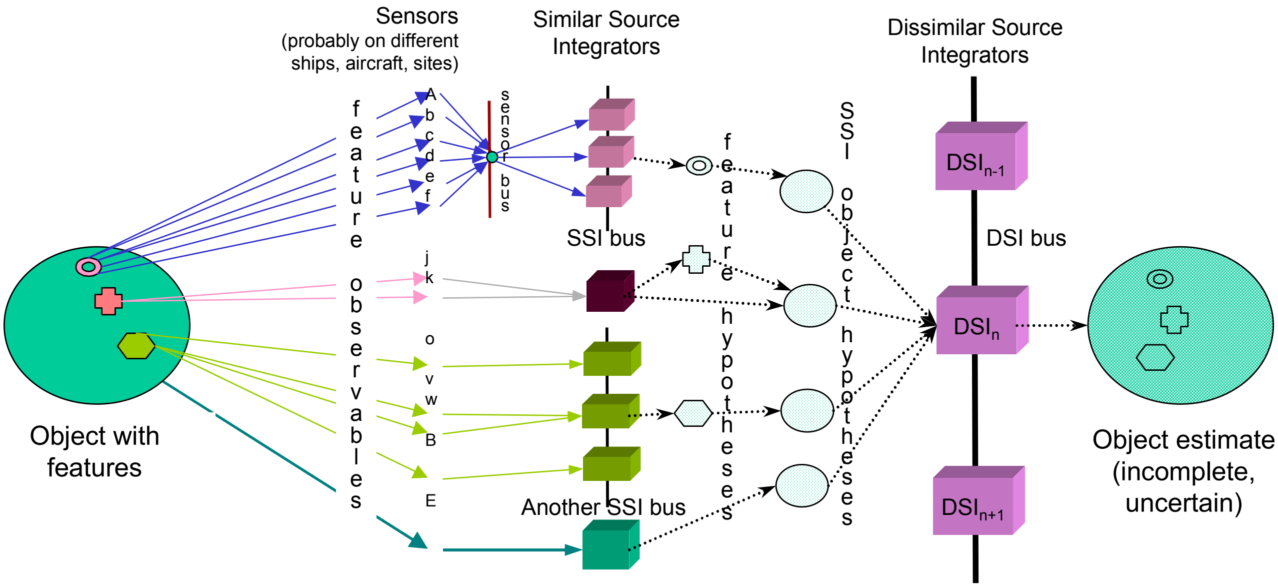


**Figure 3. Level 1 Processing Flow**

Notional level 2 processing flow is shown in Figure 4. In this flow, the left object is a complex battlefield object such as a Division or communications network. It consists of many objects such as the one operated upon in the Level 1 process. Each of these individual objects has observables that pass through the level 1 processing to product individual object estimates and hypotheses. In some cases, an observable regarding the composite object is received, such as from COMINT internals. The multiple object estimates and hypotheses, along with the direct composite object measurements are processed by Multi-Source Integration (MSI) functions. The MSI's collaborate on an MSI bus, such as the COP bus. The result is estimates and hypotheses regarding the complex battlefield object.

The level 3 process is shown in Figure 5. The complex battlefield estimates and hypotheses generated from the level 2 process are used to generate alternative predictions of future action and states. These state hypotheses are consistent with the current estimates and hypotheses. As in the level 2 process, some measurements directly indicate future hypotheses, e.g., from COMINT internals.

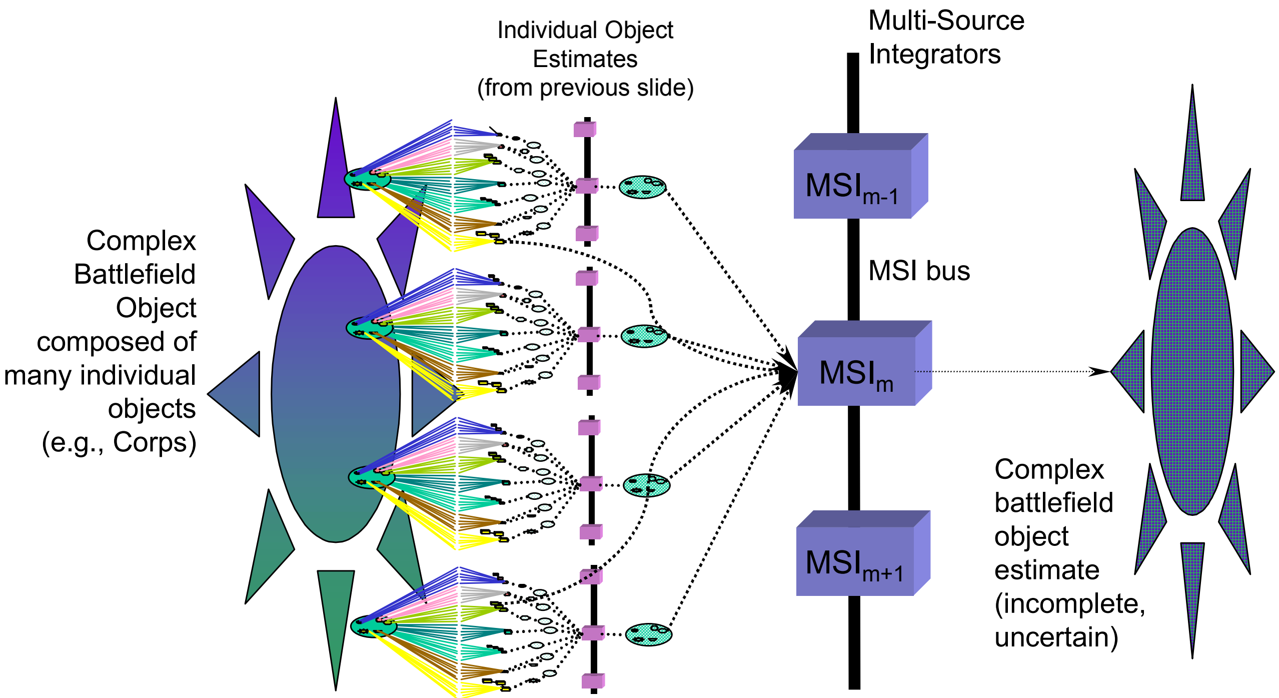


**Figure 4. Level 2 Fusion Process Flow**

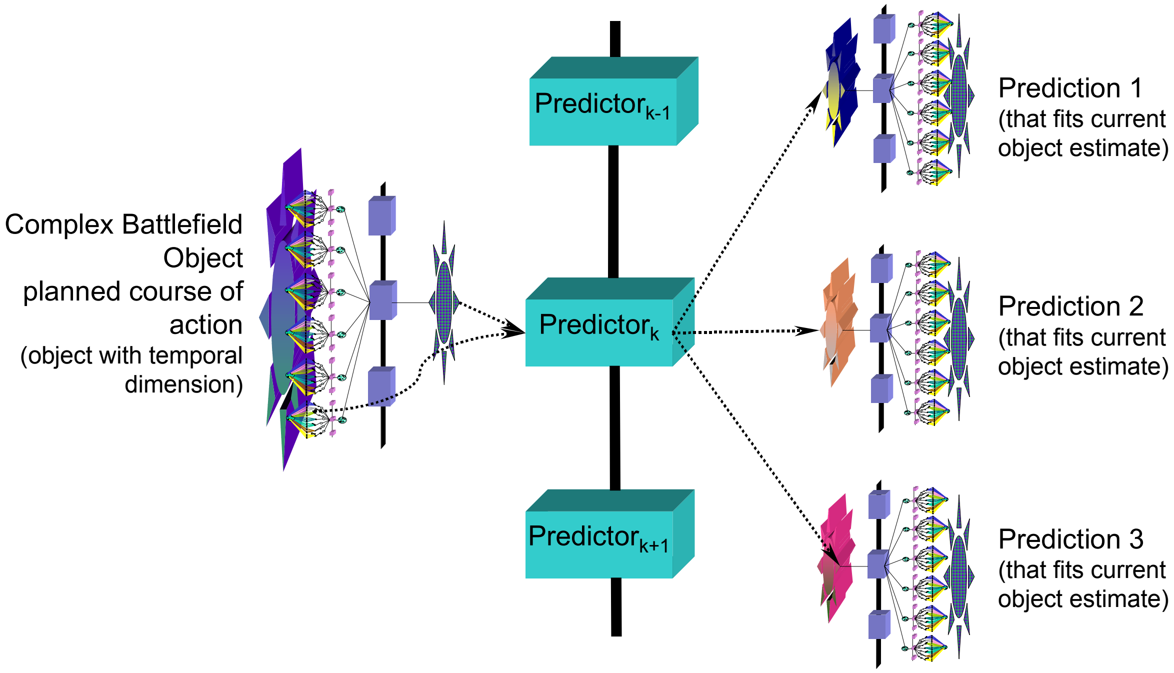


**Figure 5. Level 3 Fusion Process Flow**

## 2.1 General Techniques for Information Fusion

For levels 1-3 fusion, techniques can be categorized as follows[4]:

- Complementary Composition
- Multi-Input Refinement
- Cross-Information Inference
- Information Requirements Analysis
- Negative Information Inference

These are shown in Figure 6 and described in the following subparagraphs.

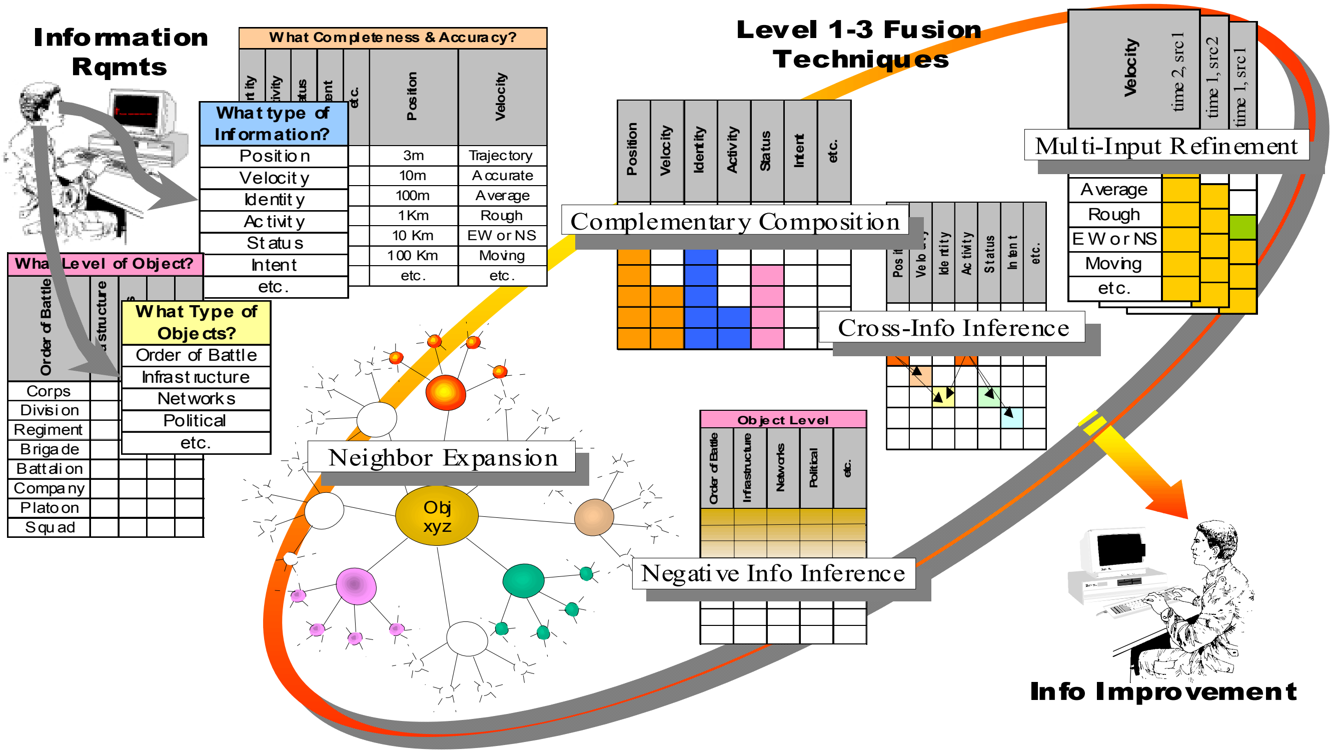


**Figure 6. General Fusion Techniques for Levels 1-3**

- Complementary composition, shown in Figure 7, refers to assembling information types measured by different sources into a composite object. For example, radar measurements can be used to derive accurate and complete kinematics while ELINT measurements can be used to derive detailed target identification.

[4] Joint C4ISR Decision Support Center (DSC) FY 2000 Study Task 2, *Multi-INT Fusion Performance*

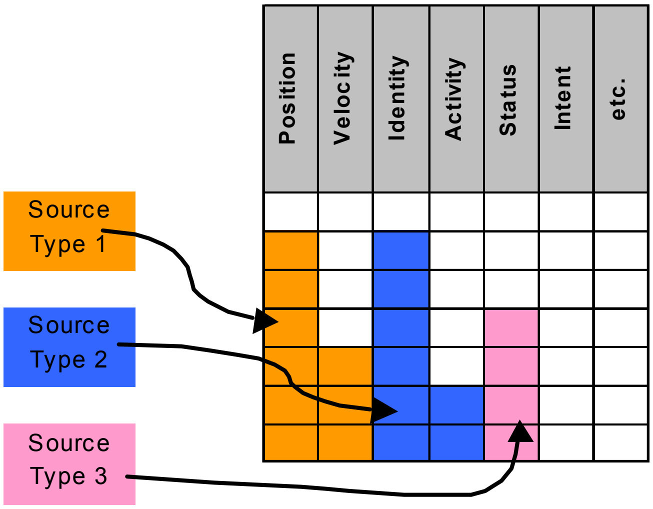


Figure 7. Complementary Composition

- Multi-input refinement, shown in Figure 8, refers to the technique by which successive inputs, applied correctly, improve data quality. This is a generalization of the statistical fact that multiple samples reduce the error bound of an estimate. Commonly applied to target kinematics, done correctly the compounding of evidence will increase the accuracy of other information types as well.

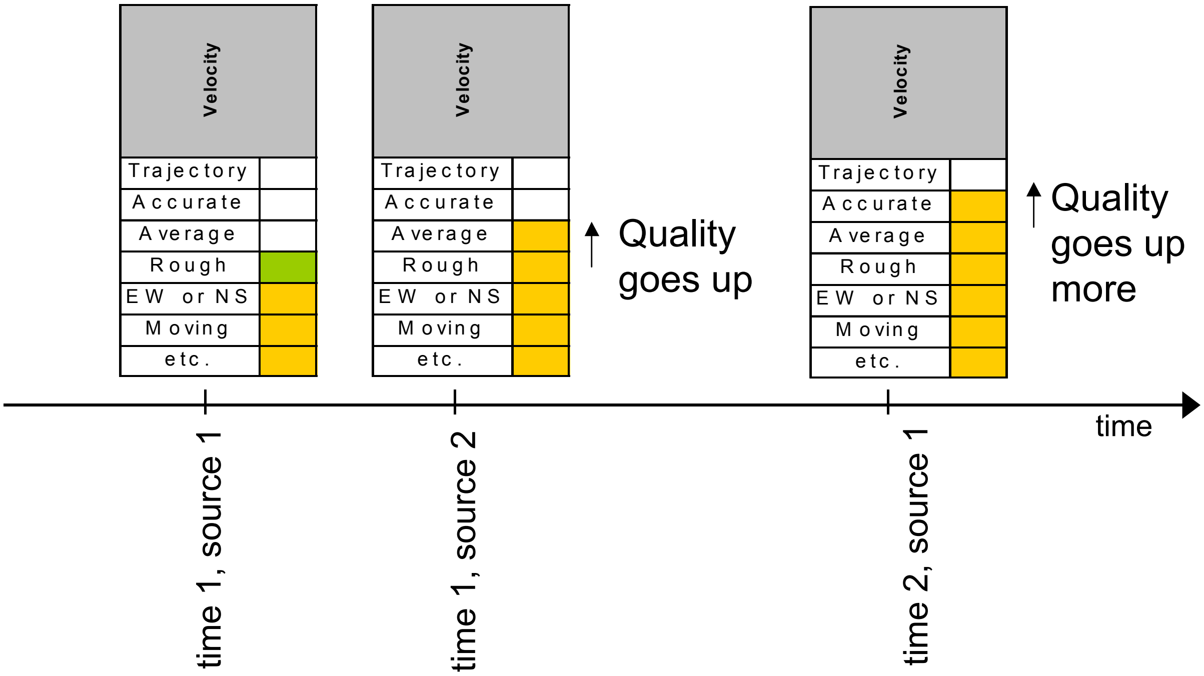


Figure 8. Multi-Input Refinement

- Cross information-type inference, shown in Figure 9, refers to the ability to infer one information type from another. Examples are, inferring velocity from successive inputs of position, inferring status (e.g., dead) from activity (e.g., none), and inferring intent (e.g., planning for attack) from activity (e.g., mobilization).

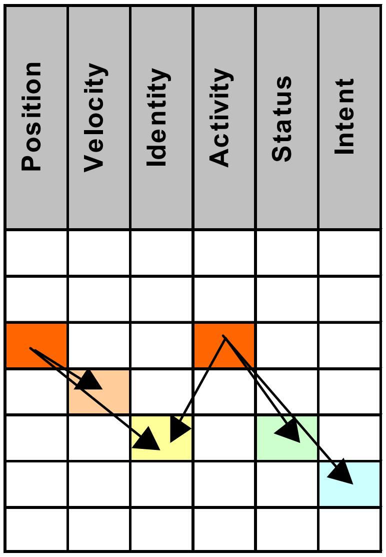


Figure 9. Cross Information-Type Inference

- Neighbor expansion, shown in Figure 10 and Figure 11, refers to the ability to infer information about related objects. For example, estimating a Brigade's center of mass from knowledge of the member Regiments, or a Battle Group from the individual ships, or a missile launcher from the missile.

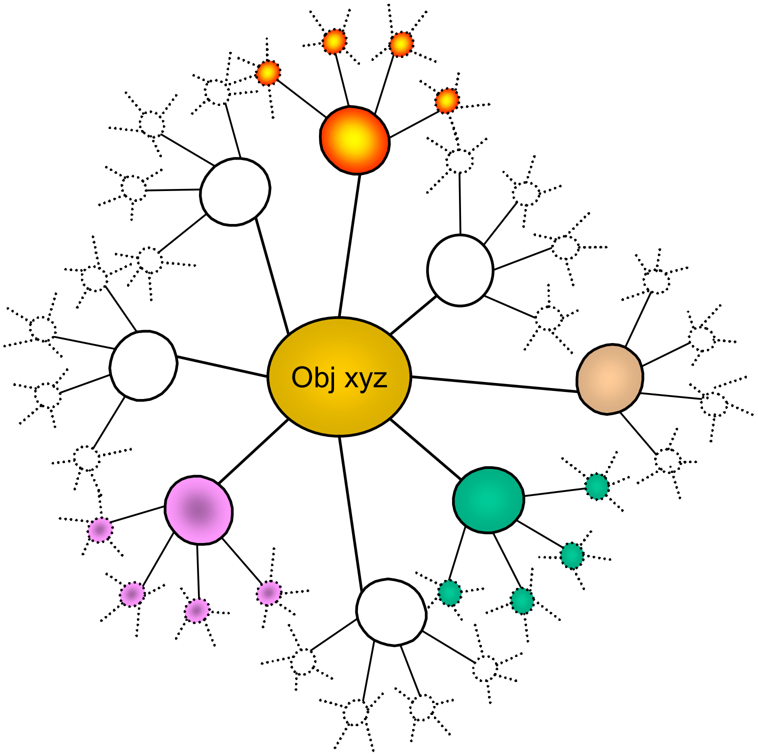


Figure 10. Neighbor Expansion Concept

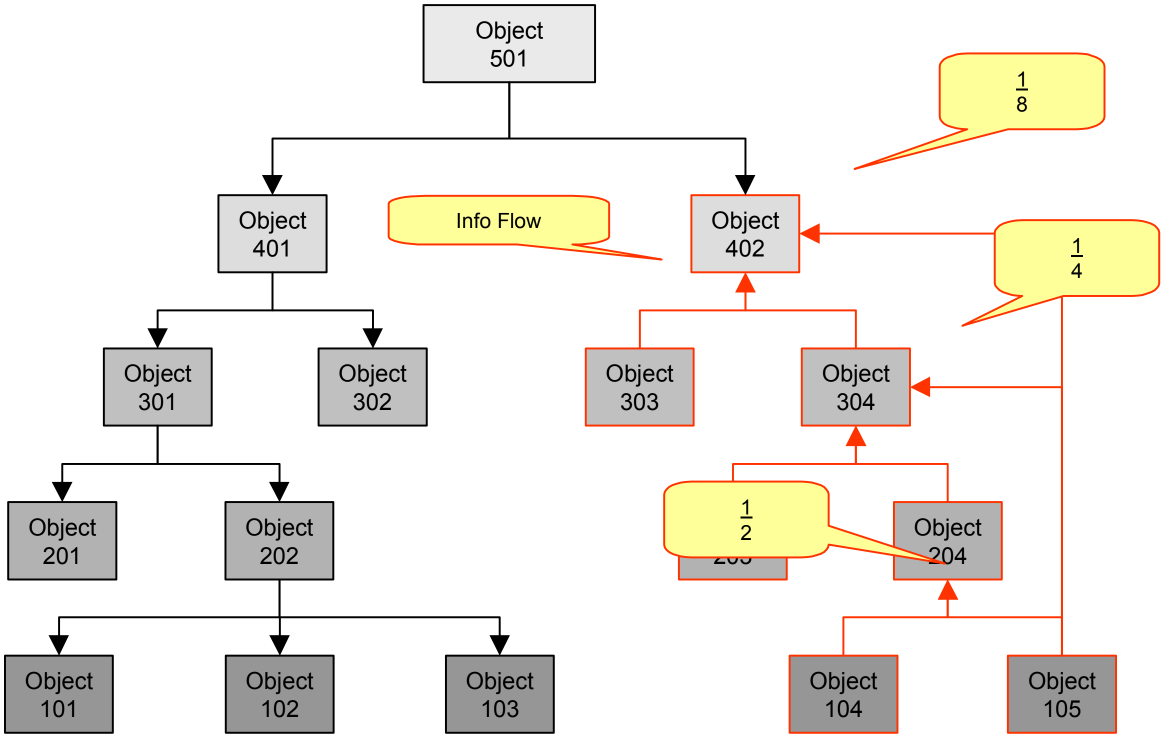


**Figure 11. Neighbor Expansion Notional Example**

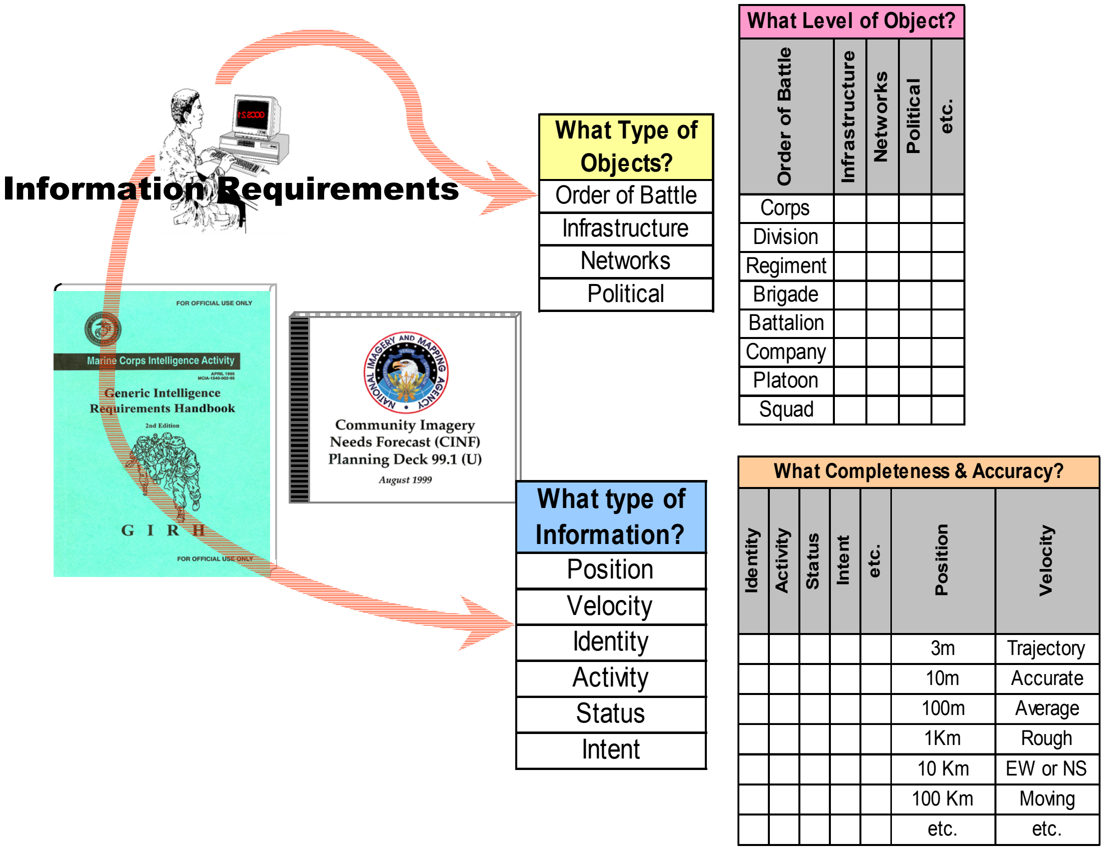
- Negative Inference refers to the fact that knowing "not" tells something about "what". For example, the UAV, Predator, sees nothing in an area means enemy is not there and is, therefore, more likely everywhere else. Negative inference can provide valuable Situation Awareness information.

## 2.2 Information Requirements Analysis

For the DSC study, analyzed several sources* and determined information types and detail and object types and level that were required

- Army Tactical Needs Database (ATNDB)
- Assured Support to Operational Commanders (ASOC) (1998)
- Commanders' Information Needs Assessment (CINA)
- Community Information Needs Forecast (CINF)
- Generic Information Requirements Handbook (GIRH) (USMC)
- US Forces Korea list provided to J-2

Based on these validated sources, the information requirements were categorized and characterized as shown in Figure 12. The categories and characterizations were not pre-determined but were derived from the information requirements.



What Level of Object?

| | Order of Battle | Infrastructure | Networks | Political | etc. |
|---|---|---|---|---|---|
| Corps | | | | | |
| Division | | | | | |
| Regiment | | | | | |
| Brigade | | | | | |
| Battalion | | | | | |
| Company | | | | | |
| Platoon | | | | | |
| Squad | | | | | |

What Completeness & Accuracy?

| Identity | Activity | Status | Intent | etc. | Position | Velocity |
|---|---|---|---|---|---|---|
| | | | | | 3m | Trajectory |
| | | | | | 10m | Accurate |
| | | | | | 100m | Average |
| | | | | | 1Km | Rough |
| | | | | | 10 Km | EW or NS |
| | | | | | 100 Km | Moving |
| | | | | | etc. | etc. |

**Figure 12. Information Requirements Analysis for the Multi-INT Fusion Study**

## 2.3 Generalized Version of the Fusion Levels for Data Integration

With only minor modification, the fusion levels can be adjusted for data integration:

a. Level One Fusion Processing – Object Refinement. Level one processing combines parametric data from multiple ~~sensors~~ sources to determine the ~~position, kinematics,~~ state and other attributes ~~or identity~~ of low level entities.

b. Level Two Fusion Processing – Situation Refinement. Level two processing develops a description or interpretation of the current relationships among objects and events in the context of the operational environment. The results of this processing is a determination or refinement of the ~~battle/~~operational situations.

c. Level Three Fusion Processing – ~~Threat~~ Strategic Refinement. Level three processing develops ~~a threat~~ an extra-organizational oriented perspective of the data to estimate ~~enemy~~ extra-organizational capabilities, identify ~~threat~~ opportunities, estimate ~~enemy~~ extra-organizational intent, and determine levels of ~~danger~~ .risk.

d. Level Four Fusion Processing – Process Refinement. Level four processing monitors and evaluates the ongoing fusion process to refine the process itself, and guides the acquisition of data to achieve optimal results. These interactions among the data function levels and with external systems or the operator to accomplish their purpose.

# 3 Data Integration Challenges

There are many challenges in integrating data from multiple disparate sources. Figure 13 simplistically illustrates the fundamental challenge categories. First, the data source must be accessible. This is the focus of most data integration activity today, from intranets, to DBMS format, to XML. Second, there must be translatable semantics. Data semantics can be rigorously modeled like all semantics using semantic nets, although the specific notation of entity-relationship modeling is generally used. Another popular style is object-oriented modeling. Finally, once the data is accessed and completely understood, multi-source differences in assertions must be reconciled. As Figure 13 illustrates, the data integration problem is not specific to DBMS's or even computer science, but is a general problem in all information exchange.

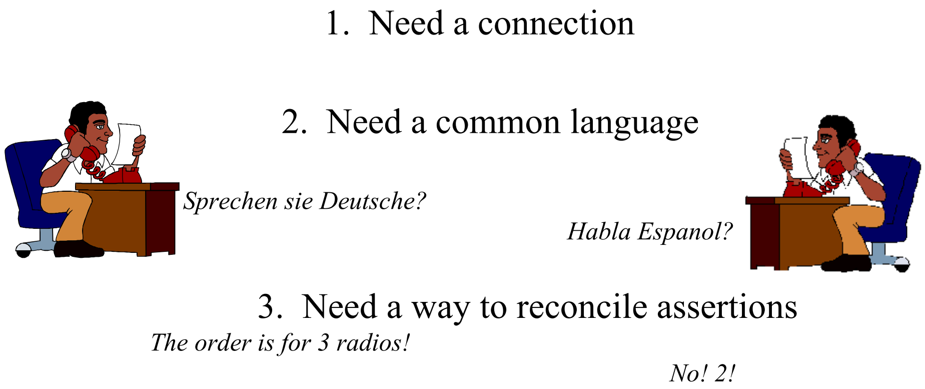

1. Need a connection
2. Need a common language
Sprechen sie Deutsche?
Habla Espanol?
3. Need a way to reconcile assertions
The order is for 3 radios!
No! 2!

**Figure 13. Data Integration Problem**

## 3.1 The Tractable Problems

The access problem in data integration, while receiving most attention today, can be considered a relatively tractable problem in that there are many known solutions that solve the problem completely. Data access problems are problems of affordable bandwidth, security policy, and access format standardization. They tend to fall into the OSI protocol layers 1-6:

1. Connectivity (physical, datalink, network, transport)
2. many solutions
3. Data access (session)
4. ODBC, DBMS's, XML, virtual databases, ETL tools
5. Data format (presentation)
6. XML, virtual databases, metadata managers, ETL tools, DBMS's

## 3.2 The Hard Problem

Hard problems are not really problems but more accurately facts of life that must be coped with in that there is no "solution". Semantics and reconciliation are such hard problems. There is and never will be a solution to the "problems" of semantics and differences of belief – they have been part of human activity since the dawn of history. What science does for these hard problems is find ways to deal with them better. In data integration, the hard problems manifest themselves as:

- Different Domain Values

- Different Identifiers and Labels
- Different Structures
- Different Definitions
- Conflicts and Evidence Pooling

*Attributes & Membership*
*Entity Relationship*
*Generalization*

*Different object levels*
*Object intersection*
*Different measurement sources*
*Indirect measurements*

Examples of Types of Problems are provided in the following subparagraphs.

## 3.3 Domain Values

Figure 14 shows an example of differing domain values for a very common data element, Friend or Foe. Three of the sets are from DoD standard data elements; the fourth is from another DoD standard, the standard for command and control systems, TADIL-J. The second example, in Figure 15, is from two DoD standard data elements. Translation tables can be built to specify, for example, what is to be done with Suspect when interfaced to a data source that does not have Suspect and vice-versa. However, the translations can only be accurate most of the time; there are times they are wrong and will produce unintended results.

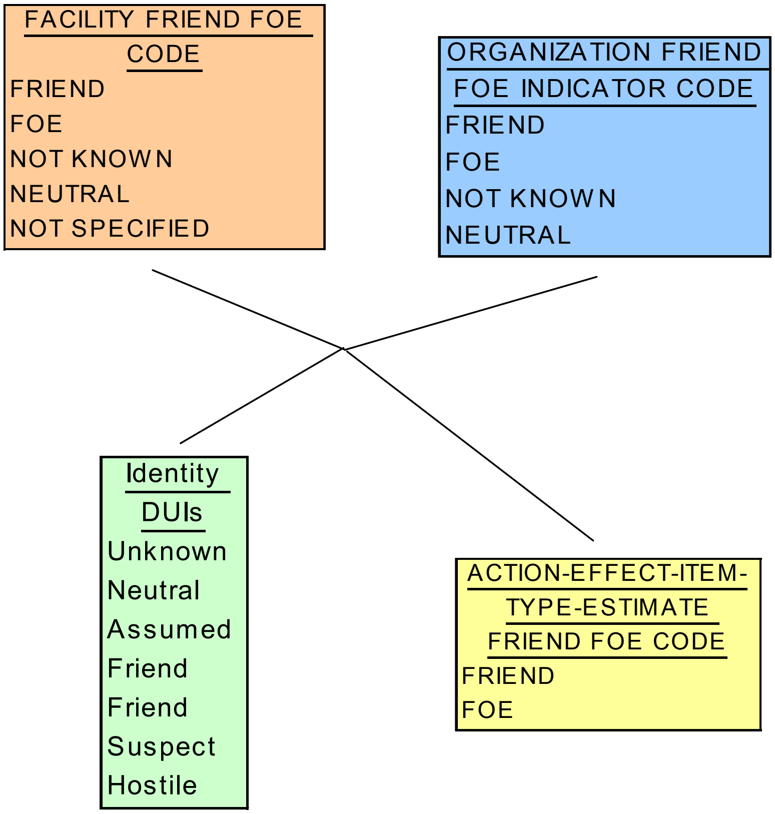


**Figure 14. Example of Differing Domain Values for Same Attribute/Field/Column**

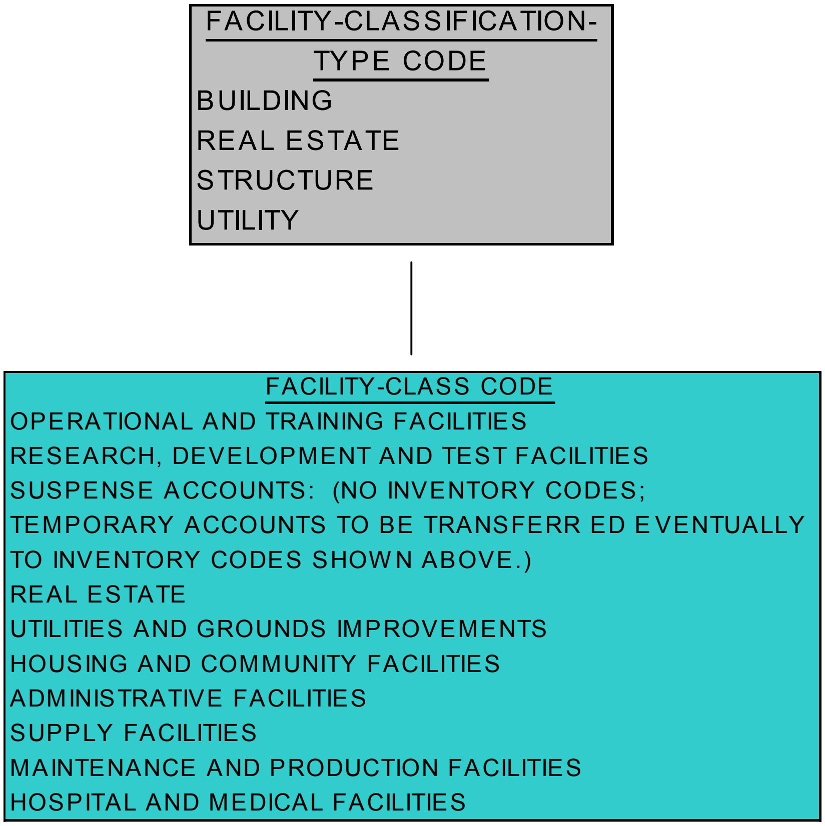

FACILITY-CLASSIFICATION-TYPE CODE
BUILDING
REAL ESTATE
STRUCTURE
UTILITY
FACILITY-CLASS CODE
OPERATIONAL AND TRAINING FACILITIES
RESEARCH, DEVELOPMENT AND TEST FACILITIES
SUSPENSE ACCOUNTS: (NO INVENTORY CODES;
TEMPORARY ACCOUNTS TO BE TRANSFERR ED EVENTUALLY
TO INVENTORY CODES SHOWN ABOVE.)
REAL ESTATE
UTILITIES AND GROUNDS IMPROVEMENTS
HOUSING AND COMMUNITY FACILITIES
ADMINISTRATIVE FACILITIES
SUPPLY FACILITIES
MAINTENANCE AND PRODUCTION FACILITIES
HOSPITAL AND MEDICAL FACILITIES

**Figure 15. Example Domain Value Differences for Same Attribute/Field**

## 3.4 Identifiers and Labels

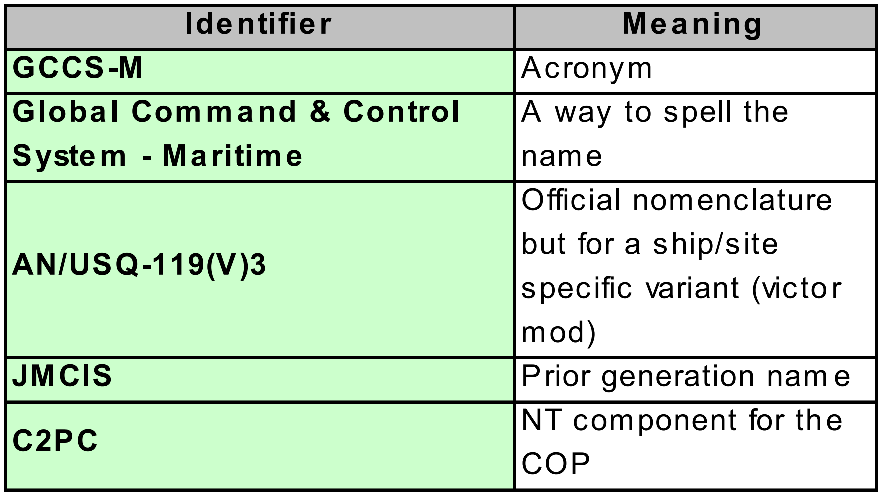
Figure 16 shows an example of different identifiers or labels for the same object, in this case the aircraft carrier USS John F. Kennedy. Hull numbers, nicknames, different punctuations, and then special identification codes in the intelligence community all refer to this ship. A simple translation table can solve this type of problem but building and maintaining complete, accurate, and validated tables can be costly and is rarely done. Even for large objects such as ships, a worldwide database of naval and

| Identifier | DB's Using |
|---|---|
| CV-67 | various |
| JFK | various |
| Kennedy | various |
| USS John F. Kennedy | various |
| U.S.S. John F Kennedy | various |
| USS Kennedy | various |
| CV 67 | various |
| BE number | NID, Intel reports |
| DIA equpment code | MIDB |
| UIC | Admin msgs |

**Figure 16. Example of Diverse Labels and Identifiers for Same Object**

merchant vessels exceeding 150 ft. requires tens of thousands of translations. The Subject Matter Expertise to maintain these is not plentiful.

Figure 17 is a more complex example because the object being referenced may not be the same; it is ambiguous knowing just the identifier. Some of the differences and ambiguities may or may not be significant, depending on the application. For example, for some applications, reference to the prior generation or current may be inconsequential but not in others. Similarly, the ship/site variant may or may not be significant. Translations between these cannot be done with simple translation tables but require models of the systems, their genealogy and planned evolution, the variants, and the components in addition to the synonymous identifiers.



| Identifier | Meaning |
|---|---|
| **GCCS-M** | Acronym |
| **Global Command & Control System - Maritime** | A way to spell the name |
| **AN/USQ-119(V)3** | Official nomenclature but for a ship/site specific variant (victor mod) |
| **JMCIS** | Prior generation name |
| **C2PC** | NT component for the COP |

**Figure 17. Harder Example of Diverse Labels and Identifiers for Same Object**

## 3.5 Structure

Three categories of structure differences are challenging to multi-source data integration, as described in the following subparagraphs.

### 3.5.1 Attribute Membership

An entity representing the same object can have very different attributes, as shown in the example in Figure 18. Depending on the application the data source is supporting, very different attributes of the object may be modeled.

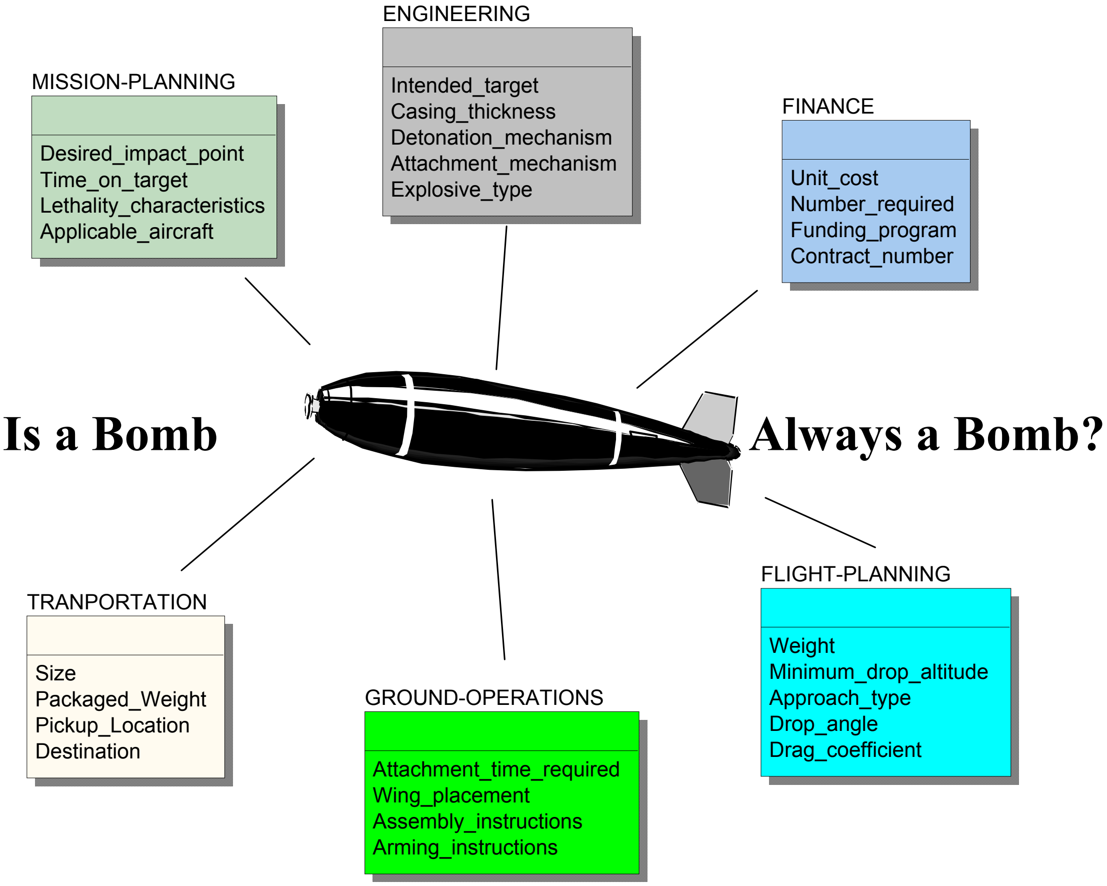


**Figure 18. Example of Differing Attributes for "Same" Object**[5]

### 3.5.1.1.1 Entity Relationship

Figure 20 is an example of differing entity-relationship modeling. In this example, the source is not third normal while the target is. In order to integrate data from the source to the destination, associative entity instances will have to be created and type codes set. This will result in a many-to-many translation.

### 3.5.1.1.2 Generalization

Enterprise-level models such as the Defense Data Architecture models are not only third normal, but also highly generalized, employing "strong typing" to accomplish what is sometimes called Universal Data Modeling. An example is the Command and Control Core data model, the DoD standard for operational data. A conceptual depiction is shown in Figure 19. In this model, the entity, "MATERIEL", has strong typing so that it is the head of a class hierarchy that ultimately covers everything from aircraft carriers to paper clips. Without strong typing, thousands of entities would be required instead of the 323 in the current model. Similar to the prior examples of integrating non-normalized data sources with normalized ones, integrating non-generalized sources with generalized ones involves complex many-to-many translations.

[5] Adapted from MITRE Corporation, Bedford MA

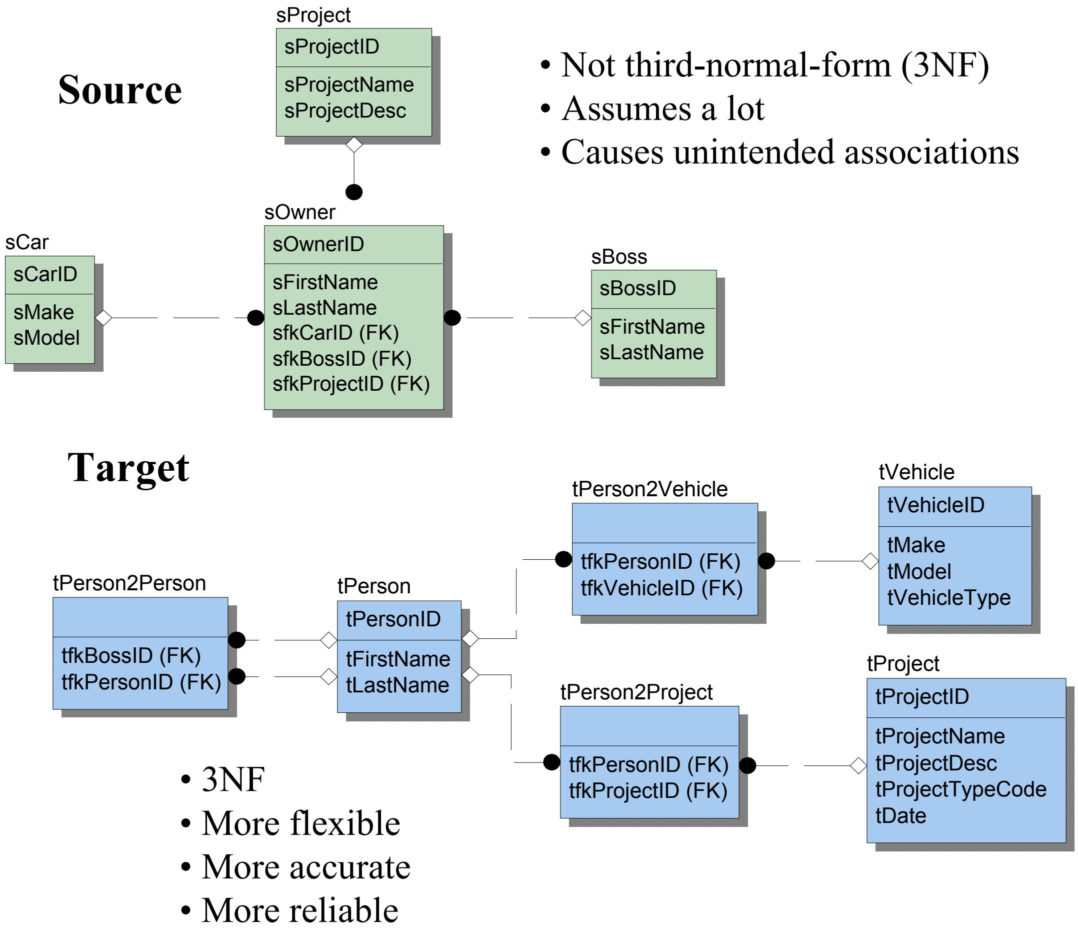


**Figure 20. Example of Different Data Abstraction and Normalization Styles**

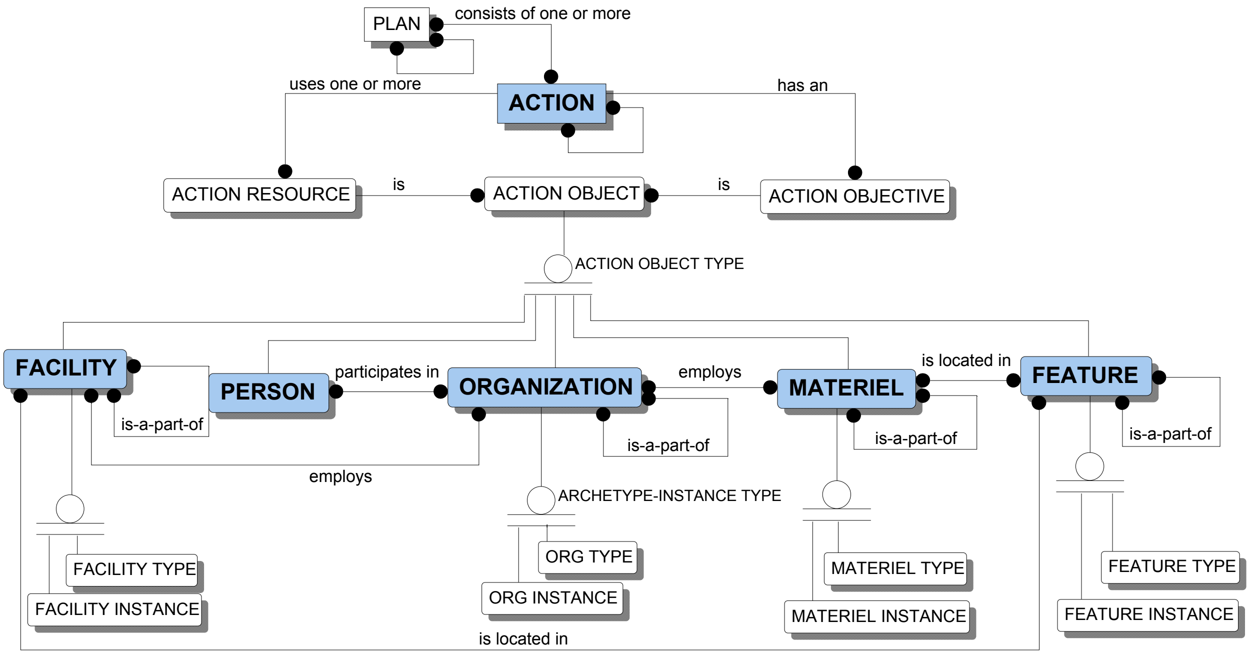


**Figure 19. Example of Highly Generalized Data Modeling**

## 3.6 Definitions

Entities apparently representing the same object may actually represent different objects. Figure 21 shows some examples. In the Ship Type example datasource 2 is wrong by most Naval definitions; however, this has occurred in a Navy database.

## 3.7 Conflicts and Evidence Pooling

There are many challenges in reconciling the different fact assertions in multiple data sources.

### 3.7.1 Different levels of aggregation

Figure 22 shows examples of different levels of aggregation for the same object class. Information from different data sources pertaining to the different levels of aggregation can be difficult, if not impossible, to integrate to the lower levels of aggregation.

### 3.7.2 Object Intersections

A more challenging variation of the prior examples of different levels of aggregation is intersections over the object class. Examples are shown in Figure 23. In the first example, datasource 1 contains data about a class of ships, while datasource 2 contains data about a combat system suite that is applicable to subsets of two classes.

**Example: Ship Type**

| *Datasource 1* | *Datasource 2* |
|---|---|
| CVN, CG, DDG | Nimitz, Ticonderoga, Los Angeles |

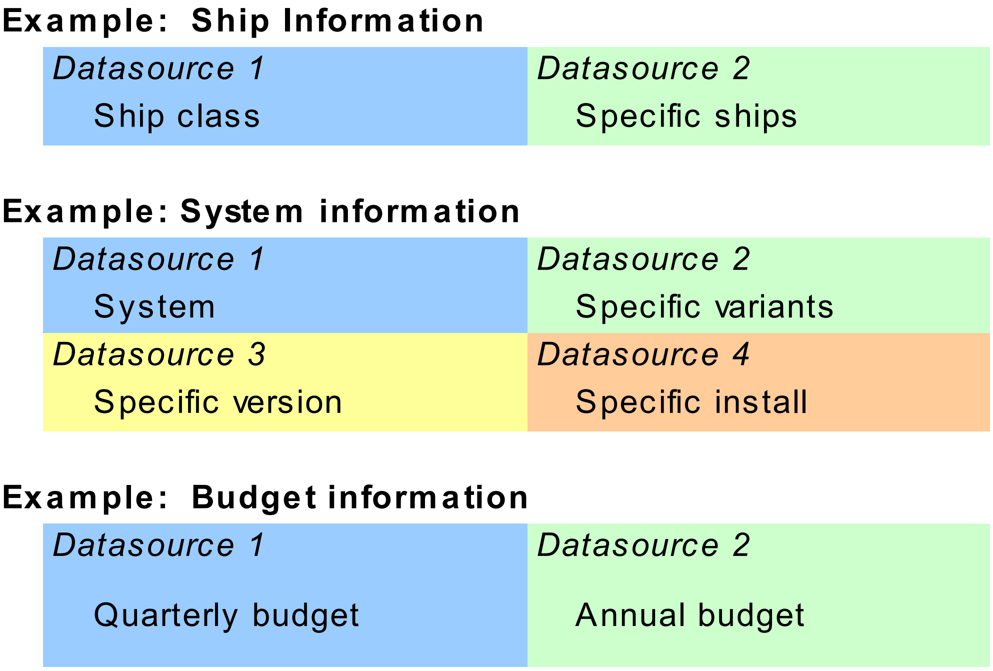
**Example: OPFAC**

| *Datasource 1* | *Datasource 2* |
|---|---|
| Buildings | Organizational units |

**Example: Platform**

| *Datasource 1* | *Datasource 2* |
|---|---|
| AWACS, DDG-51 | 386, powerpc, RISC, 680x0 |

**Figure 21. Examples of Differing Meanings for Same-Titled Entity**



**Example: Ship Information**

| | |
|---|---|
| *Datasource 1*<br>Ship class | *Datasource 2*<br>Specific ships |

**Example: System information**

| | |
|---|---|
| *Datasource 1*<br>System | *Datasource 2*<br>Specific variants |
| *Datasource 3*<br>Specific version | *Datasource 4*<br>Specific install |

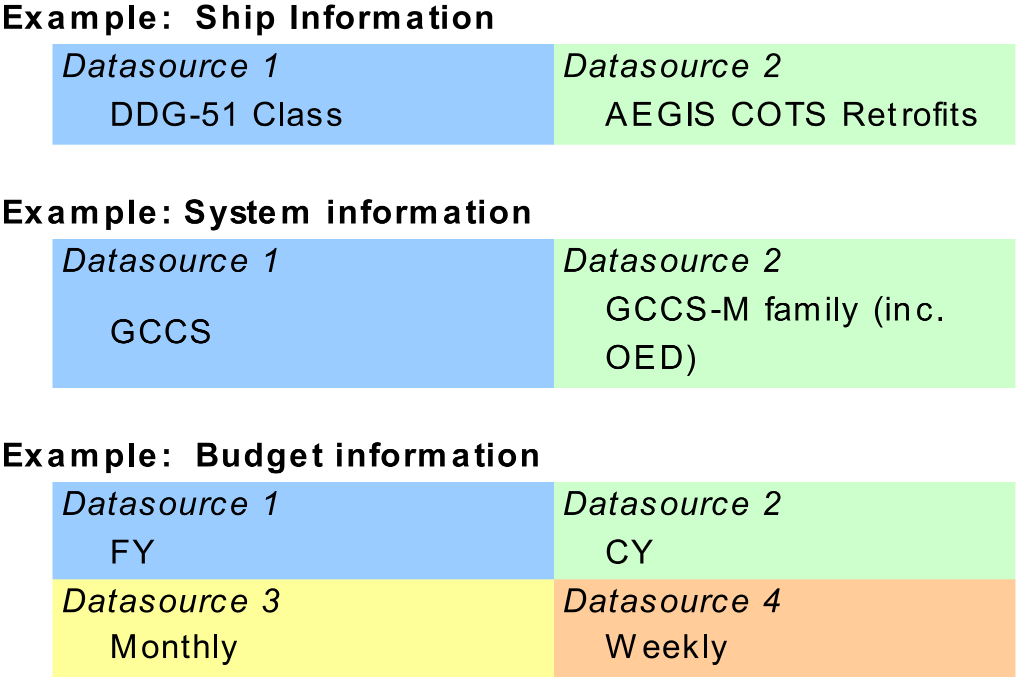
**Example: Budget information**

| | |
|---|---|
| *Datasource 1*<br>Quarterly budget | *Datasource 2*<br>Annual budget |

**Figure 22. Examples of Differing Levels of Same Object**

### 3.7.3 Different measurement sources

Different input, or measurement sources, can assert different information about the same object. Examples are shown in Figure 24. In the first example, the ship installation data, even if at the same level of detail, has different authority depending on the timeline. In the second example, the data can vary by time, interpretation, belief, source contacted, etc.



**Example: Ship Information**

| | |
|---|---|
| *Datasource 1*<br>DDG-51 Class | *Datasource 2*<br>AEGIS COTS Retrofits |

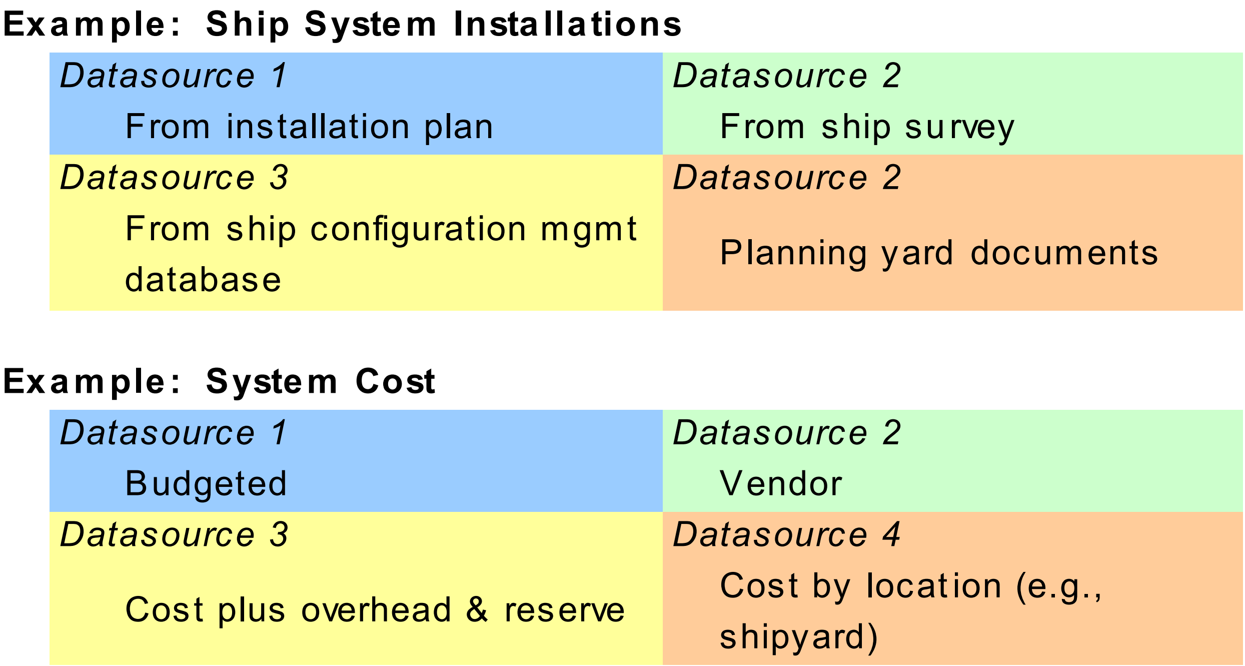
**Example: System information**

| | |
|---|---|
| *Datasource 1*<br>GCCS | *Datasource 2*<br>GCCS-M family (inc. OED) |

**Example: Budget information**

| | |
|---|---|
| *Datasource 1*<br>FY | *Datasource 2*<br>CY |
| *Datasource 3*<br>Monthly | *Datasource 4*<br>Weekly |

**Figure 23. Example of Object Intersections**



**Example: Ship System Installations**

| | |
|---|---|
| *Datasource 1*<br>From installation plan | *Datasource 2*<br>From ship survey |
| *Datasource 3*<br>From ship configuration mgmt database | *Datasource 2*<br>Planning yard documents |

**Example: System Cost**

| | |
|---|---|
| *Datasource 1*<br>Budgeted | *Datasource 2*<br>Vendor |
| *Datasource 3*<br>Cost plus overhead & reserve | *Datasource 4*<br>Cost by location (e.g., shipyard) |

**Figure 24. Examples of Different Measurement Sources for Same Assertion**

### 3.7.4 Indirect measurements

Often data in a datasource is derived from data from other sources, whether automatically, by import, or by re-keying. In some cases, the multi-source inputs could be better than single source as a result of selection and synthesis of more data points. On the other hand, the selection and/or synthesis could be wrong or the rules could be out-of-date. Other factors that influence the validity of derived data are refresh periodicity (how often is selection / synthesis revisited), variable validity times that could degrade selection / synthesis, source latency that could degrade indirect assertions, and continuity of connection to source(s).

# 4 Applying Data Fusion Techniques to Data Integration

This section discusses how some of the fusion techniques described in section 2 could be applied to solve some of the problems described in section 3. The fusion techniques that appear to be applicable are Level 1 Similar Source Integration (SSI) and Dissimilar Source Integration (DSI), Level 2 Multi-Source Integration, Level 3 Predictor, and overall information requirements analysis techniques. These appear directly applicable to the problems of

- Different Domain Values
- Different Identifiers and Labels
- Different Structures
- Different Definitions
- Conflicts and Evidence Pooling

Improvements with these problems will then lead to improved:

- Object Knowledge Improvement

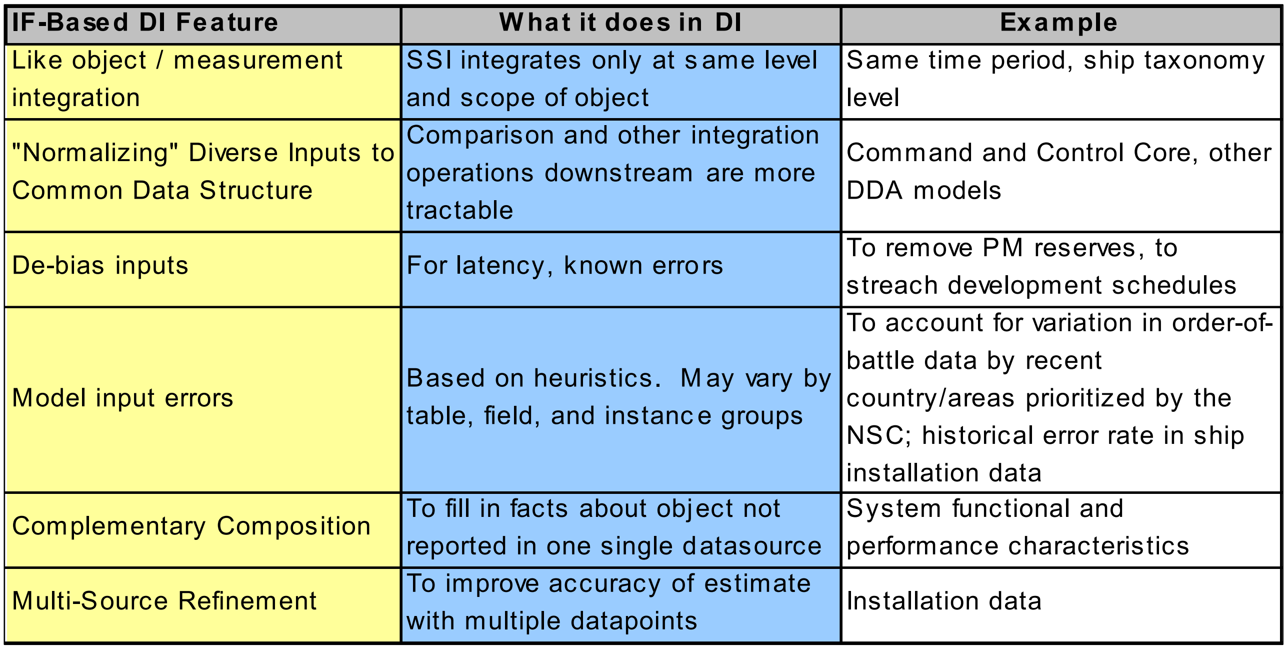
- Situation Awareness
- Strategic Assessment

## 4.1 Level 1 Similar Source Integration Techniques

Similar Source Integration (SSI) is defined as the function that

> "…associates information on common objects from similar sensors within and amongst sites, and develops platform tracks and identification estimates based on this associated information. The SSI function provides the necessary capabilities to individually correlate radar, ELINT and COMINT, and other specialized data into tracks for immediate use by the Command functions and for passing to the Organic Dissimilar Source Integration function. Individual SSI system functions shall have the capability to exchange information among cooperating platforms via Contact/Parametric Nets. This function includes data management necessary for correlation, tracking, and identification."[6]

SSI applies to data integration as shown in Figure 25.



| IF-Based DI Feature | What it does in DI | Example |
|---|---|---|
| Like object / measurement integration | SSI integrates only at same level and scope of object | Same time period, ship taxonomy level |
| "Normalizing" Diverse Inputs to Common Data Structure | Comparison and other integration operations downstream are more tractable | Command and Control Core, other DDA models |
| De-bias inputs | For latency, known errors | To remove PM reserves, to streach development schedules |
| Model input errors | Based on heuristics. May vary by table, field, and instance groups | To account for variation in order-of-battle data by recent country/areas prioritized by the NSC; historical error rate in ship installation data |
| Complementary Composition | To fill in facts about object not reported in one single datasource | System functional and performance characteristics |
| Multi-Source Refinement | To improve accuracy of estimate with multiple datapoints | Installation data |

**Figure 25. Level 1 Similar Source Integration Fusion Techniques Applicable to Data Integration**

- Like object / measurement integration. The role of the SSI level of fusion is to fuse like measurements first. For example, there are ESM/ELINT and Radar SSI's. Applied to data integration, it would integrate only at same level and scope of object. For example, at the SSI level data would be integrated only across the same levels of aggregation such as same time period or ship taxonomy level. Diverse aggregation or period integrations then occur downstream in the Dissimilar Source Integration function.
- "Normalizing" Diverse Inputs to Common Data Structure. This is separation of concern technique that is employed in many data fusion systems. Input sensor or intelligence data is translated to a

[6] Navy C4ISR System Architecture, SPAWAR, 1998

standard, or normalized, format. Then all downstream fusion functions and algorithms can operate on a single data structure, thus streamlining the fusion functions. For data integration applications, translation to a common reference model enables complementary composition and adjudication of multi-source inputs. The common reference model should employ data abstraction techniques so that it is capable of handling a wide variety of inputs in an integrated manner. Examples of such common reference models are Command and Control Core and the other DDA models.

- De-bias inputs. Fusion systems calibrate or register input sensor data to a common measurement reference using specific calibration tests or estimating bias over input time. Examples of biases estimated are North misalignment (for radars), navigation error, and time latency. Examples in the data integration domain could be removal of known cost buffers (reserves) and extension of development schedules.

- Model input errors. For fusion applications, it is typical to model the sensor or intelligence source to infer input data characteristics such as error bounds, typically not provided in the input data stream. For data integration applications, these may be based on heuristics that indicate the quality, authority, or error bounds of input data. These may vary by table, field, and instance groups. Examples of use in data integration are accounting for variation in order-of-battle data by recent country/areas prioritized by the NSC and historical error rate in ship installation data.

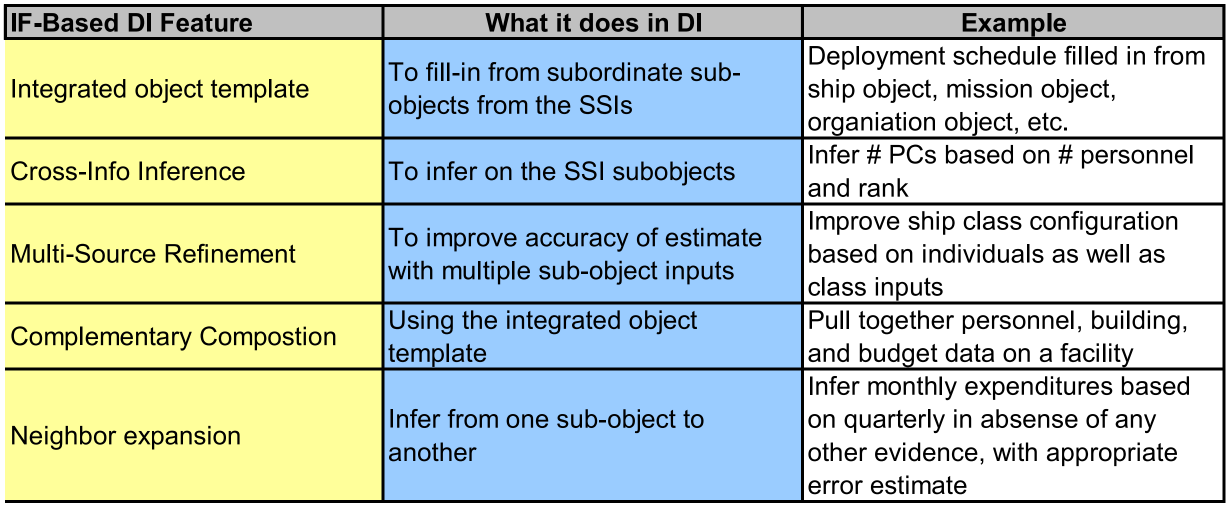
- Complementary Composition. Early uses in data fusion applications were augmenting 2-D radar data with height-finding data to form a 3-D position. The purpose is to fill in facts about object not reported in one single datasource. For data integration applications this enables are more complete estimate of the object(s) of interest base on individual sources that contain partial data.

- Multi-Source Refinement. Data fusion systems treat multi-source inputs as statistical samples from which statistically merged results can be derived that have greater accuracy than the individual samples. The measurement process is statistically ergodic so that time-series inputs can be treated as samples. Once the source data qualities (error bounds) are modeled and the business rules for fusion are developed, the same principal applies to data integration.

## 4.2 Level 1 Dissimilar Source Integration Techniques

Dissimilar Source Integration (DSI) is the function that,

> "… provides for data fusion of force organic sensor information and the sharing of this information with Organic Dissimilar Source Integration processors on other platforms."



| IF-Based DI Feature | What it does in DI | Example |
|---|---|---|
| Integrated object template | To fill-in from subordinate sub-objects from the SSIs | Deployment schedule filled in from ship object, mission object, organiation object, etc. |
| Cross-Info Inference | To infer on the SSI subobjects | Infer # PCs based on # personnel and rank |
| Multi-Source Refinement | To improve accuracy of estimate with multiple sub-object inputs | Improve ship class configuration based on individuals as well as class inputs |
| Complementary Compostion | Using the integrated object template | Pull together personnel, building, and budget data on a facility |
| Neighbor expansion | Infer from one sub-object to another | Infer monthly expenditures based on quarterly in absense of any other evidence, with appropriate error estimate |

**Figure 26. Level 1 Fusion Dissimilar Source Integration Techniques Applicable to Data Integration**

The applicability of DSI techniques to data integration is shown in Figure 26.

- Integrated object template. DSI's integrated object template allows the particular object features estimated by the diverse SSI's to be applied in an integrated manner. Examples in the data integration domain could be deployment schedule filled in from ship object or mission object filled in with the organization object.

- Cross-Information Inference. DSI's can infer information that is not directly measured based upon known relationships between information. For example, target activity can be inferred from movement or radar operating mode. In data integration applications, an example is inferring the number of computers at a site based on the number of personnel and occupational specialties. Inferences are always estimates and DSI's maintain an estimate of accuracy with all inferences. In some cases the error bound can be quite large, but it is almost always less than complete ignorance.

- Multi-Source Refinement. Analogous to the SSI's, DSI's improve the accuracy of estimates with multiple sub-object inputs. For example, DSI refines the estimate of a SAM site based on inputs regarding the location of launchers, command trailers, and various radars. In data integration application, a ship class configuration could be refined based on individual ship as well as class inputs.

- Complementary Composition. DSI's use the integrated object template to juxtapose information regarding the estimated sub-objects or phenomenologies of the object of interest. An example in a data integration application would pull together personnel, building, and budget data on a facility.

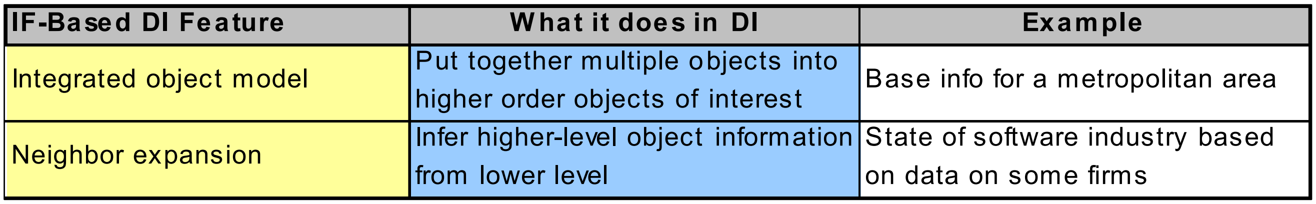
- Neighbor expansion. DSI's use neighbor expansion to infer information about an unmeasured object based on measurements of related objects. For example, the location of a missile launcher can be inferred from the measured missile. An example in data integration would be inferring monthly expenditures based on quarterly in absence of any other evidence, with appropriate error estimate.

## 4.3 Level 2 Multi-Source Integration Techniques

The Multi-Source Integration (MSI) function:

> "…consists of Wide-Area Surveillance (WAS) and tracking of space resources. WAS provides fusion of information from National and Coast Guard wide-area sensors and sharing of information with Dissimilar Source Integration processors on other platforms or at shore."

MSI estimates information about composite objects such as Divisions, Battle Groups, and electrical networks, often based on measurements of the component objects. MSI techniques applicable to data integration are shown in Figure 27.



| IF-Based DI Feature | What it does in DI | Example |
|---|---|---|
| Integrated object model | Put together multiple objects into higher order objects of interest | Base info for a metropolitan area |
| Neighbor expansion | Infer higher-level object information from lower level | State of software industry based on data on some firms |

**Figure 27. Level 2 Fusion Multi-Source Integration Techniques Applicable to Data Integration**

- Integrated object model. An essential foundation for MSI is an integrated object model that describes how individual objects compose or are aggregated into higher level objects. For example, how battalions form into regiments or how subnets form into communications networks. The same types of models are used in data integration applications to tie lower object data with higher object data and to come to conclusions about higher level objects.

- Neighbor expansion. Neighbor expansion is the MSI technique that uses the integrated object model to infer information between levels of objects. In data integration applications this would nudge the confidence of a hypothesis across object levels. An example would be inferring the state of software industry based on data on some firms (or vice-versa.)

## 4.4 Level 3 Predictor Techniques

Level 3 fusion formulates and estimates the probability of various courses of action. Level 3 fusion techniques applicable to data integration are shown in Figure 28.

| IF-Based DI Feature | What it does in DI | Example |
|---|---|---|
| Integrated temoral model | To fil-in alternative predictions as "ghosts" of the current estimate | Alternative DoD budgets |
| Neighbor expansion | Infer possible future states from current | Trends based on historical and other evidence |
| Negative inference | Monitoring for events are not occuring | Bankruptcies of defense contractors |

**Figure 28. Level 3 Fusion Techniques Applicable to Data Integration**

- Integrated temporal model. As described previously, fusion processes operate upon an integrated object and composite object model. For level 3 fusion, the model is extended in the temporal dimension. Level 3 fusion uses this dimension to fill-in alternative predictions as "ghosts" of the current estimate. An example in the data integration domain would be alternative situation hypotheses for POM budget planning, commonly called “gaming”, or alternative courses of action for business competitors.

- Neighbor expansion. For predictions, neighbor expansion activates and/or perturbates the confidences or probability of neighboring situation hypotheses based upon the current situation hypotheses. In data integration applications, an example would be trends analysis based on historical and other evidence.

- Negative inference. Negative inference eliminates some alternative courses of action. For example, lack of troop movement or presence in certain areas could mean the enemy is not planning an approach in those areas. An example in data integration would be the lack of bankruptcies of defense contractors, suggesting greater confidence in the health of the defense contracting business.

## 4.5 Information Requirements Analysis Techniques

As described previously, an important part of any fusion system design is the analysis and characterization of the mission information requirements. Techniques used in data fusion that are applicable to data integration are shown in Figure 29.

- By responsibility and activity/task. The information requirements can be determined by the needs for what information, for what purpose, and with what characteristics. An example in data integration would be the Comptroller needs for POM's for budget submission, on a certain date, and with certain accuracy.
- By information type and detail, object type and level. The information requirements should be stated in a uniform characterization that unambiguously describes the information needed.

| IF-Based DI Feature | What it does in DI | Example |
|---|---|---|
| By responsibility and activity/task | Determine who needs what information, for what purpose, and with what characteristics | Comptroller needs POMs for budget submission, on a certain date, and with certain accuracy |
| By information type & detail, object type & level | Uniform charactierization of information requirements | What constitues the information needed |

**Figure 29. Fusion Information Requirements Analysis Techniques Applicable to Data Integration**

## 5 Summary and Conclusion

This paper described data fusion paradigms and techniques and showed how they could be generalized to data integration problems. This leads to a model for data integration based on estimation and integrated models, as shown in Figure 30.

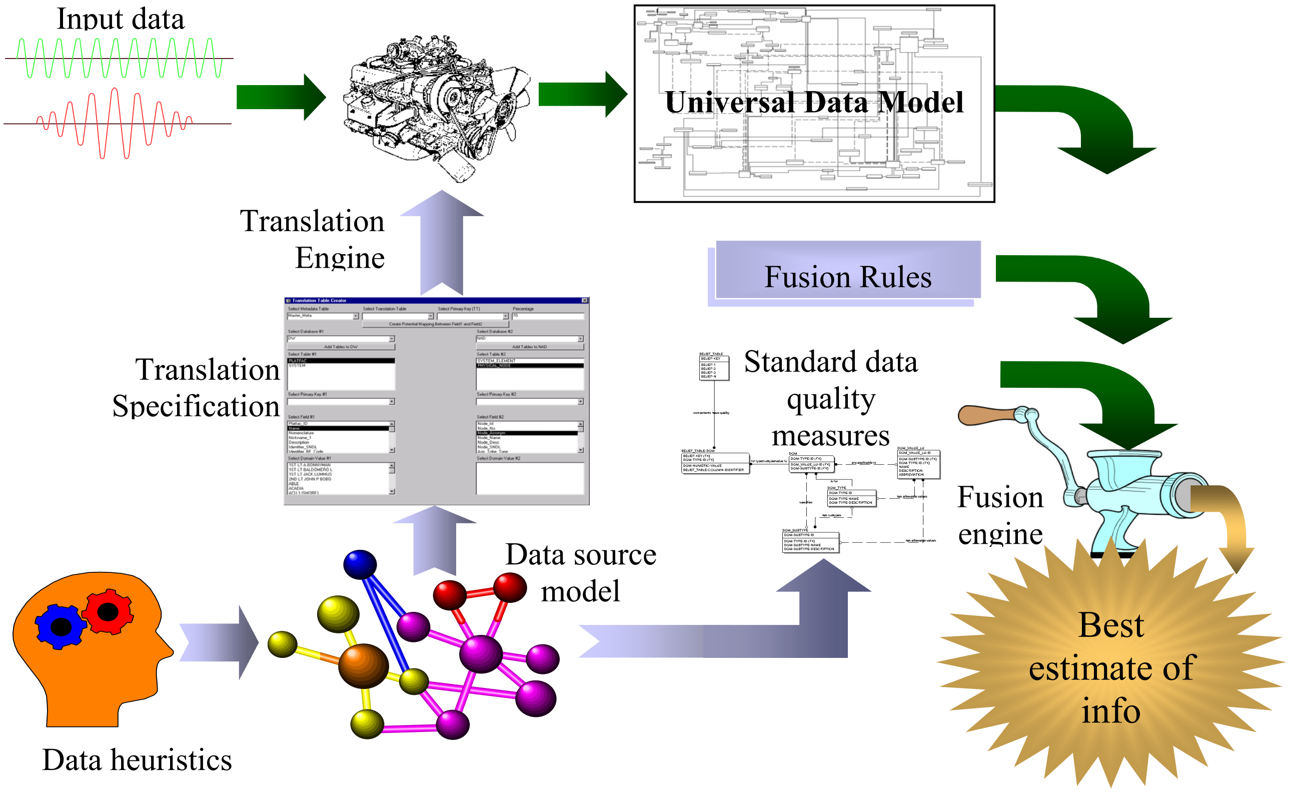


**Figure 30. Data Integration Based on Data Fusion Notional Process Flow**

This model provides a framework for further investigations and developments for data integration. Examples of on-going efforts in the Navy are the use of common integrated reference models (SSI, DSI, MSI) and information-level translation (SSI). In information-level translation, data administrators specify translation at information, rather than data level. This supports "normalizing" to highly generalized common integrated reference models from diverse sources.

# 6 Glossary

| | |
|---|---|
| AWACS | Airborne Warning and Control System |
| BE | Broad Ecension |
| C2PC | Command and Control Personal Computer |
| CG | Cruiser, Guided missile |
| COTS | Commercial Off the Shelf |
| DBMS | Data Base Management System |
| DDA | Defense Data Architecture |
| DDG | Destroyer, Guided missile |
| DI | Data Integration |
| DIA | Defense Intelligence Agency |
| DSC | Decision Support Center |
| DSI | Dissimilar Source Integrator |
| DUI | Data Unit Identifier |
| ELINT | Electronics Intelligence |
| EO | Electro-Optical |
| ETL | Extraction, Transformation, and Loading |
| EW | Electronic Warfare |
| IF | Information Fusion |
| INTEL | Intelligence |
| JCTN | Joint Composite Tracking Network |
| JDL | Joint Directors of Laboratories |
| JMCIS | Joint Maritime Command Information System |
| MIDB | Modernized Integrated Data Base |
| MSI | Multi-Source Integrator |
| NID | Naval Intelligence Dataset |
| ODBC | Open Data Base Connectivity |
| OED | Ocean surveillance intelligence system Evolutionary Development |
| OPFAC | OPerational FACility |
| OSI | Open Systems Interconnection |
| PM | Program Manager |
| SIGINT | Signals Intelligence |
| SSI | Similar Source Integrator |
| UAV | Unmanned Aerial Vehicle |
| UIC | Unit Identification Code |
| XML | eXtensible Markup Language |