Presented at the 5th International Conference on Information Fusion

Multi-Hypothesis Database for Large-Scale Data Fusion

Dave McDaniel
Silver Bullet Solutions, Inc.
4747 Morena Blvd., Suite 350
San Diego, CA 92117
davem@silverbulletinc.com

Abstract -- *Progress in deploying large-scale fusion systems has been slow in recent years despite substantial algorithmic developments. One reason is that there has not been a way to address a large-scale enterprise in a tractable manner that allows modular and collaborative evolution of fusion algorithms. Information and data modeling techniques have become quite mature over the past 20 years so that it is now possible to model the information domain of a large-scale enterprise tractably. By extending information modeling constructs to semantic and inference nets, it is possible to use these information models as a basis for large-scale fusion. This paper shows how to instrument an information model into a fusion inference structure. Algorithms encapsulation and computing techniques are discussed. This approach could lead to foundations for large-scale fusion in defense, intelligence, law enforcement, and air traffic control systems.*

Keywords: Entity-Relationship Modeling, Semantic Network, Inference Network, Data Fusion

1 Introduction

Under a general definition of inference, sensor and data fusion can be looked at as inference processes, particularly at fusion levels 2 and 3 [1]. Inference processes are often structured around the concept of inference networks [2]. Inference networks have similarities to semantic networks, a common representational technique for onto logic modeling and semantic analysis [3]. The concept of semantic network has similarities to semantic data modeling [4] the general method of data modeling, of which entity-relationship (E-R) modeling is a well-known technique [5]. Powerful tools have been developed for E-R modeling, such as ERwin [6] and all major Data Base Management Systems (DBMS) contain E-R modeling tools, for example, in Oracle, “Designer”, or in Sybase, “PowerDesigner”. Almost all-even modestly complex database design today employs the E-R modeling technique. Object-oriented data models can be partially modeled using E-R techniques [7].

The success of the E-R technique has led to the development of large and highly expressive global-view models that cover vast enterprises. For instance, within the US Department of Defense (DoD), over 250 standard enterprise models have been developed containing over 30,000 data elements [8]. These cover every activity of the department, from finance and personnel to sensor signals and weapons employment, in an integrated manner.

This paper describes an approach for using these types of E-R models as a foundation for fusion inference networks. Structural and computational issues are discussed. The enabling features of E-R models, modern DBMS's, and object-oriented techniques are discussed. It will thus outline a way to construct very large inference nets, of the type needed in many applications such as defense, intelligence, and law enforcement.

2 Background

The role of data models in modeling concepts have been investigated for several years, for instance [9] on Semantic Data Modeling (SDM). Others ([10], 11], [12]) have investigated how conceptual modeling relates to database modeling.

Entity-Relationship (E-R) modeling is a relatively mature computer science technique for database design, pioneered by Chen [13, enriched and applied to relational database design by Codd [14] and popularized by many, particularly Date [5]. In an E-R model, an Entity is an object in the domain of interest. Entities have Attributes, which can be thought of as their inherent properties. For example, the entity, PERSON, has inherent properties such as mass and birth date. Mass may change over time, but it is inseparable from the person to whom it belongs. A property such as Employer, in contrast, is not inherent to the PERSON, but is instead a relationship between the PERSON and an ORGANIZATION. This is because most people participate in many organizations (job, societies, political organizations, churches). None of these are part of the PERSON. The success of the E-R technique is evident in its integration with leading development methodologies such as IDEF [15] and Universal Modeling Language (UML) [16] and in its incorporation in DBMS tools that provide methodical and traceable progression from conceptual models to physical models to automatic generation of databases. Figure 1 shows a screen shot from a popular tool (ERwin by Computer Associates) poised to generate a database is shown in.

Copyright 2002 © Silver Bullet Solutions, Inc.

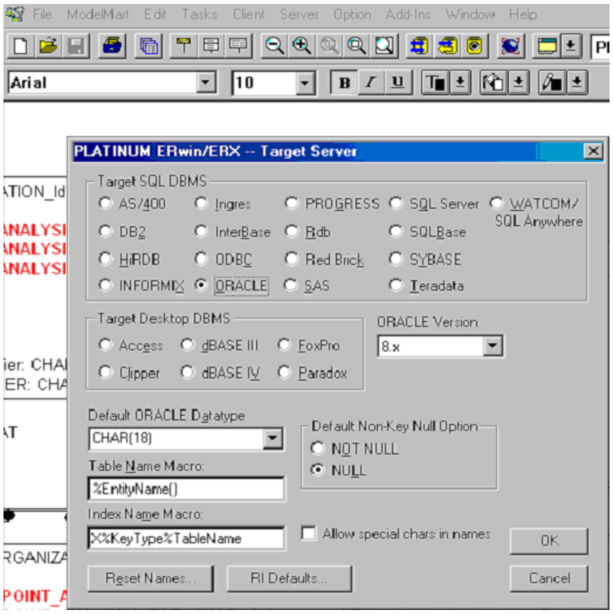


Figure 1. An E-R Modeling Tool About to Generate a Database Automatically

A different approach, but having some similar features to semantic data modeling, is the Semantic Network. They have also been investigated for some time, the first reference ascribed to Margaret Masterman in 1961 [17]. A reference on on-going research is [18]. In a semantic network, concepts are modeled as well as the spreading of propositions to confirm or reject hypotheses regarding objects of interest. The spreading is normally node-oriented, but may be path–oriented [19].

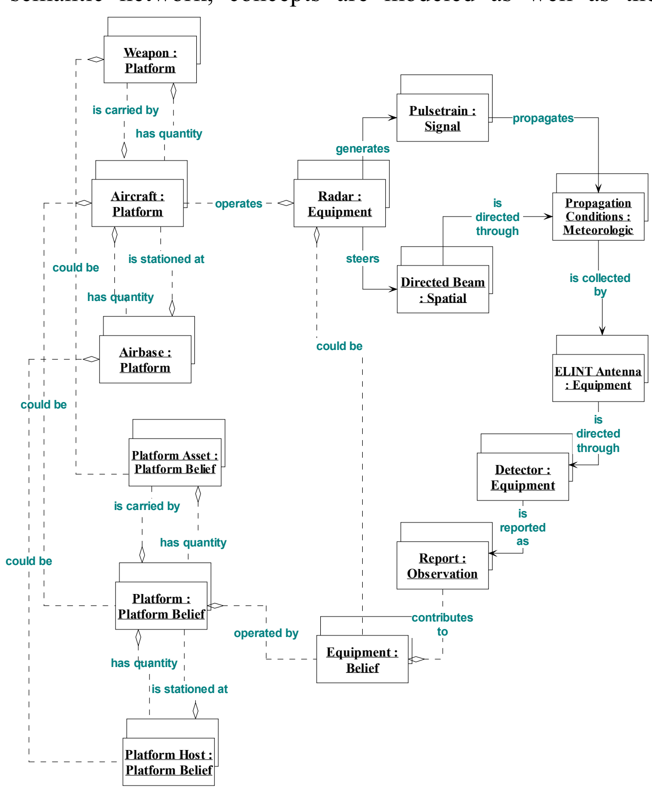


Figure 2. Example Bayesian Net for ELINT/ESM Fusion

Table 1. Multi-INT Fusion Study Information Categories

Information Requirement Category	
Kinematics	Location, velocity, and trajectory (past and predicted), from detection to accuracy sufficient for PGMs
Identification	Broad type to specific unit and with varying certainty
Activity	General to specific plan and with varying certainty
Status	General to specific and with varying certainty
Intent	General to specific and with varying certainty

Inference networks bear some similarities to semantic networks. Inference network diagrams depict the objects in the domain of interest and their inter-relationships. The objects and relationships modeled are all those that influence belief regarding the state of the primary object(s) of interest. In the example shown in Figure 2 [20], the primary objects of interests are the platforms and facilities that have radar type equipment. All objects and relationships that influence belief regarding their state (kinematics, identity, activity, in this case) are modeled.

3 Command and Control Core Data Model

The now commonly accepted definition of fusion levels [1] provides a common basis and framework for fusion researchers, system developers, and users. It also defines the scope of fusion broadly, to cover not just target kinematics, but identity, activity, and other attributes of individual targets, their organizational, mission, and other associations and groupings, the attributes of such groups, their intent and possible future states, and so on. Generalization of these levels to non-military domains has been discussed by McDaniel [21].

A recent study by the US DoD [22] analyzed the information scope for multi-intelligence fusion. In this study, an information requirements model was developed to answer the question, What are the information needs of the soldiers that might improved by alternative fusion architectures? Thousands of authoritative information requirements were analyzed and categorized as to the required information type and quality. All information requirements were found to categorize into the information categories shown in Table 1. The object types these pertained to could be categorized at a high-level as shown in Table 2.

Table 2. Multi-INT Fusion Study Object Types

Object Type	
Platforms and Facilities	Ships, aircraft, missiles, vehicles, SOF units, SAM sites, TELs, etc. from Company level up to Corps level.
Infrastructure	Communications networks, electrical networks/grids, transportation networks, etc.
Political	National organization, intent, internal conflicts, economic triggers and indicators, etc.

Copyright 2002 © Silver Bullet Solutions, Inc.

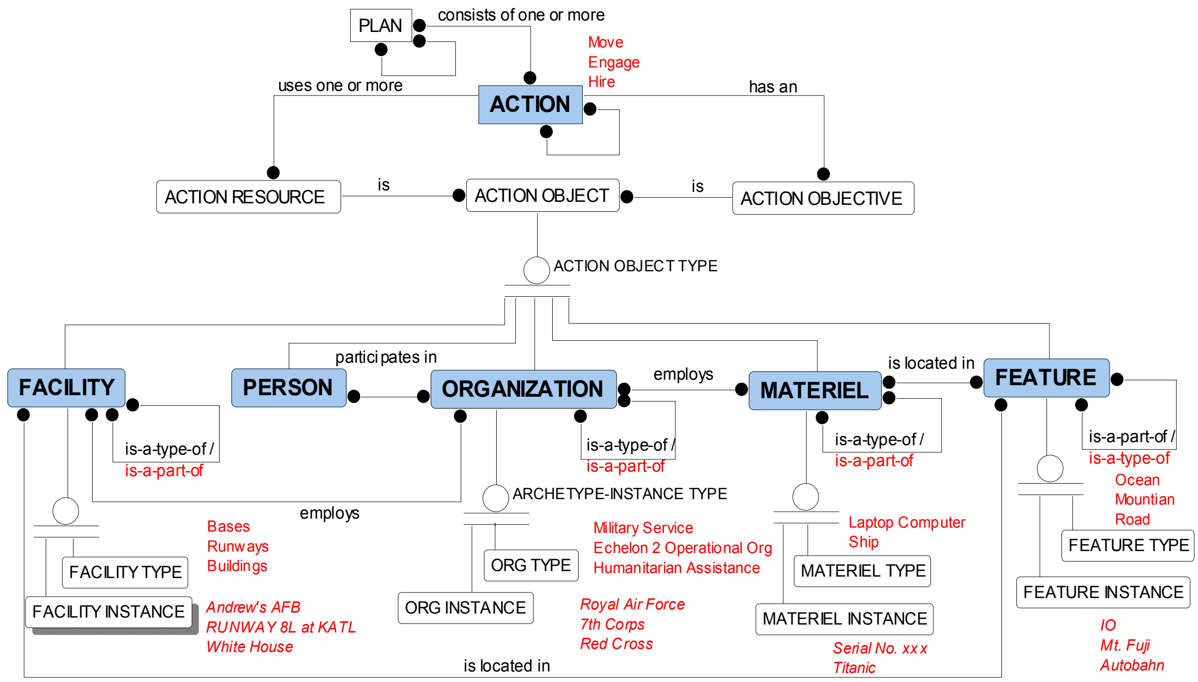


Figure 3. Major Entities in Command and Control Core Data Model

A fully attributed E-R model of much of this information is the Command and Control (C2) Core data model [8]. Figure 3 is a high-level conceptualization of C2 Core. The fully attributed model is quite large and cannot be shown herein but the miniature in Figure 4 should convey some idea of both the quantity of entities and relationships and also the high degree of integration and relatedness.

The diagram can be read as definitional sentences that state the properties between object types such as, "Action has-an Action-Objective", or, "Person participates-in Organization". These are examples of enduring properties of the objects at hand.

All of the entities shown in are what in object-oriented design would be called, "object classes", but in E-R modeling might be called abstracted or generalized entities. For instance, ACTION can be cover all types of actions, from destroying a target to submitting a budget, via the TYPE CODE and the ACTION RESOURCE's and ACTION OBJECTIVE's, either of which can be the generalized objects FACILITY, PERSON, ORGANIZATION, MATERIEL, and FEATURE, all of which are superclasses. Example instance values are shown in Figure 3, types in regular font, actual real instances in italics. The types ____TYPE carry the general class behavior for the ___INSTANCES. Subclassing types and instances under the general concept mirrors ordinary human discourse in which terms can interchangeably refer to types and instances. For example, I might speak of how fast a car can go, talking to the general type of car or a specific car. In inference systems, having a general concept as a superclass of the type and instance allows transition from classification hypotheses at the type level to hypotheses at the actual real object level.

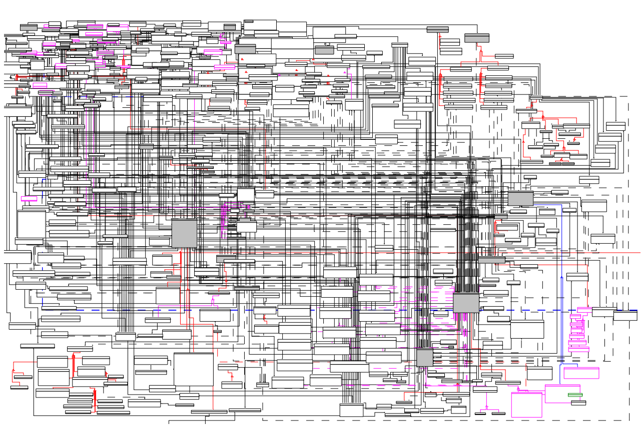


Figure 4. High Relatedness and Integration of C2 Core

Types generally have many levels of taxonomic typing. For example, targets may categorize as Air, Surface, Subsurface, Ground, Space with Air subcategorizing as Bomber, Fighter, Transport, etc. with Fighter further subcategorizing as F-15, F-16, Viggen, etc. with yet another for F-16 as F-16A, F-16B, etc. The purpose of typing is to allow for property inheritance. In inference, this will enable recognition of new instances.

4 Comparison Of E-R Models and Semantic and Inference Networks

The essential difference in E-R models and semantic networks are that abstract E-R models such as C2 Core represent the definitional level of semantics while semantic networks show assertions (see, for examples, [8].) In systems based on C2 Core, assertions show up as attribute values and associative entity instances and, if so modeled, attribute values in the associative entity. So, to say a specific armored vehicle is one nation or another in a semantic network, we might show the armored vehicle node

Copyright 2002 © Silver Bullet Solutions, Inc.

connected to Nation A or Nation B. In an abstracted database designed from C2 Core, the armored vehicle would be an instance in a table with an associative entity instance to the Nation A or Nation B instance in another table. The same information is encoded but the semantic network connections are fully explicit while in an E-R notation, only the definitions are graphically explicit.

Compound assertions are more complicated to compare. For example, Major x thinks Commander y believes Major x's Battalion is moving through the valley (author's modification of original example in [3]). Using semantic network notation, we show Major x, Think, Commander y, Believe, Battalion, Moving, and Valley as nodes with connection lines showing types of primitive semantic relationships between the nodes. For examples, see [23]. How this is handled in a database system depends on whether the database is representing Major x's thoughts or is modeling them. In the first case, the database will model Commander y's beliefs as a type of ACTION with Commander y as the ACTION-RESOURCE (the believer) and the ACTION-OBJECTIVE a pointer to the hypothesis that ACTION-RESOURCE Major x's Battalion is moving (ACTION) through the ACTION-OBJECTIVE of the LOCATION (valley). In the latter case, Major x's belief is modeled as an ACTION on pointing to Commander y's belief.

5 Instrumenting for Inference

Data models are expressive and can have great fidelity to the ontology of the modeled domain. However, they model all beliefs in binary -- true or not true. Prior work to deal with uncertainty in databases includes attribute conflict resolution in heterogeneous databases [24]. In this the Dempster-Shafer evidential technique was investigated

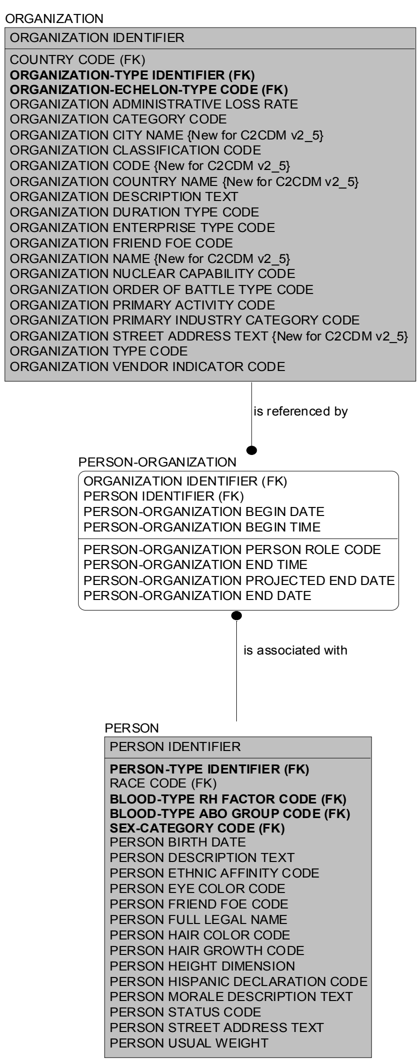


Figure 5. Example Associative Entity

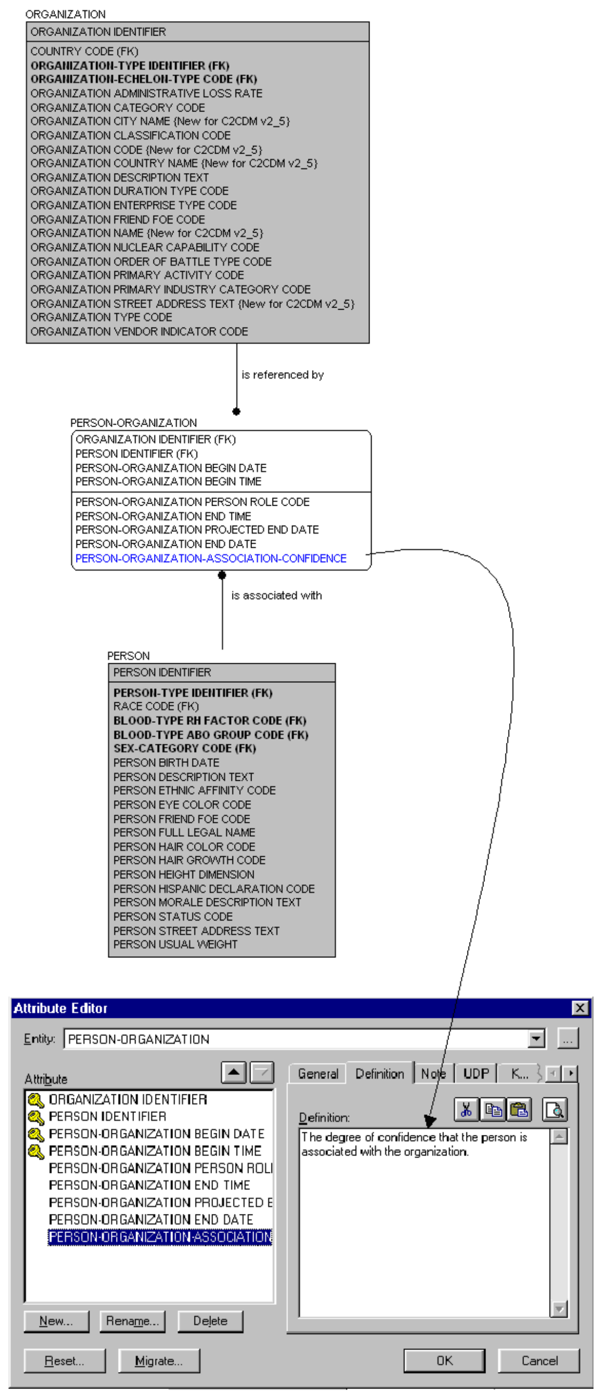


Figure 6. Instrumenting Associative Entities for Uncertainty

Copyright 2002 © Silver Bullet Solutions, Inc.

as a means to resolve conflicts in values between attributes of the same entity in different databases. Lee's [25] work in this area showed how uncertain updates could be encoded in a relational model.

There are four types of augmentations to models like C2 Core that are necessary to express uncertainty. The following will generalize from prior work to cover more types of uncertainty representation and to allow for all forms of inference, including estimation techniques.

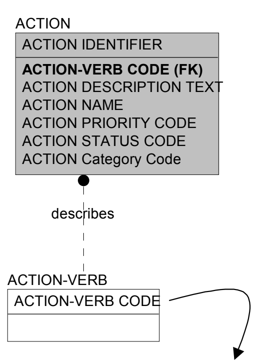
5.1 Associative Entities

The simplest case is when the model relates two objects via an associative entity, to support many-many relations and/or information about the relation that is not specific to the individual objects, as in Figure 5. This associative entity provides a many-to-many relationship between PERSON and ORGANIZATION, allowing a person to be a part of many organizations, conversely, allowing an organization to have many people. Common-sensical but powerful, it is nevertheless binary -- you're either a part or not; no maybe's. To represent uncertainty in this case, merely add a confidence value to the associative entity, as shown in Figure 6. An instance in the associative entity represents each hypothesis.

5.2 Migrated Foreign Keys

Migrated keys are keys from other entities that are part-of, or an attribute of, another entity. An example of a migrated foreign key is shown in Figure 7.

In this case an associative entity must be used in lieu of the migrated key to carry all the hypotheses and the confidence value, as shown in Figure 8. Of course, a



01--DESTROY; 02--CAPTURE; 03--NEUTRALIZE; 04--DENY; 05--MOVE; 06--AVOID; 07--INTERDICT; 08--HARASS; 09--STOP; 10--SEIZE; 11--PREPARE; 12--CLEAR; 13--SUPPLY; 14--PAUSE; 15--RESUPPLY; 16--PROVIDE SMOKE; 17--PROVIDE IMMEDIATE SMOKE; 18--SUPPRESS; 19--PROVIDE IMMEDIATE SUPPRESSION; 20--FIRE FOR EFFECT; 21--ADJUST FIRE; 22--PROVIDE PREPLANNED FIRE; 23--MARK; 24--ILLUMINATE; 25--PROVIDE CONTINUOUS ILLUMINATION; 26--PROVIDE COORDINATED ILLUMINATION; 27--PROVIDE SPREADING FIRE; 28--PROVIDE FINAL PROTECTIVE FIRE; 29--CHECK FIRE; 30--CANCEL CHECK FIRE; 31--CHECK ALL; 32--CANCEL CHECK ALL; 33--CEASE LOADING; 34--CANCEL CEASE LOADING; 35--MOVE TO CONTACT; 36--MEET AND ENGAGE; 37--ATTACK AND SEIZE; 38--CONDUCT HASTY RIVER CROSSING; 39--CONDUCT DELIBERATE RIVER CROSSING; 40--CONDUCT HASTY DEFENSE; 41--CONDUCT DELIBERATE DEFENSE; 42--WITHDRAW; 43--ASSIGN; 44--ADJUST; 45--CONTINUE; 46--CHANGE; 47--CANCEL; 48--HOLD; 49--ABORT; 50--BUILD; 51--REPAIR; 52--TRACK; 53--DETECT TARGET; 54--ACQUIRE TARGET; 55--IDENTIFY TARGET; 56--RECORD TARGET; 57--INITIATE TARGET; 58--ASSESS TARGET; 59--ENGAGE TARGET; 60--HANDOFF TARGET; 61--ACCEPT TARGET HANDOFF; 62--PROVIDE REINFORCING FIRE; 63--COLLECT TARGET INTELLIGENCE; 64--REPEAT FIRE FOR EFFECT; 65--APPROVE ACTION; 66--COORDINATE ACTION; 67--INITIATE ACTION. (DDDS, approved)

Figure 7. Migrated Foreign Key Example

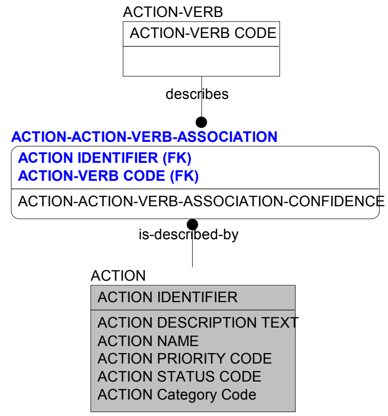


Figure 8. Embedding Uncertainty in Migrated Keys

migrated FK is just a special case of the more general associative entity, that is, one-to-many is a special case of many-to-many.

5.3 Continuous variables

The prior cases dealt with discrete variables. When continuous variable uncertainty can be represented parametrically, the additional attributes area easy, for example as in Figure 9.

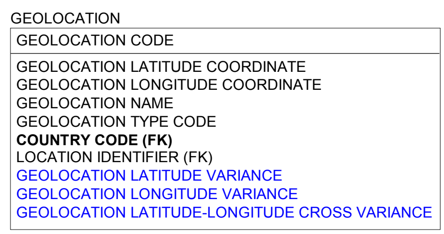


Figure 9. Parametric Case

There are cases now when it isn't possible to accurately represent uncertainty parametrically, such as due to terrain tailoring. Often, discrete PDFs are used to approximate the tailored PDF. This requires additional data structures, as shown in Figure 10.

6 Integrating State and Measurement Models

Models like C2 Core represent the state of the domain of interest, in this case the battlespace for command and control. However, most clues that effect belief in that domain are sensor measurements, not represented in C2 Core. In this case, we are in luck because the C2 Core is just a part of an overall Defense Data Architecture, all of which use common model elements. So, for example, it is fairly easy to "attach" a measurement model, such as shown in Figure 11. This one is a transmitter model, appropriate for electronics signals measurements [8]. The tie-in to C2 Core is the entity MATERIEL-ITEM. This model now completes the inference network example shown earlier in Figure 2, but far more comprehensively.

To see how this would work, imagine an update to the frequency and modulation type tables, either in the form

Copyright 2002 © Silver Bullet Solutions, Inc.

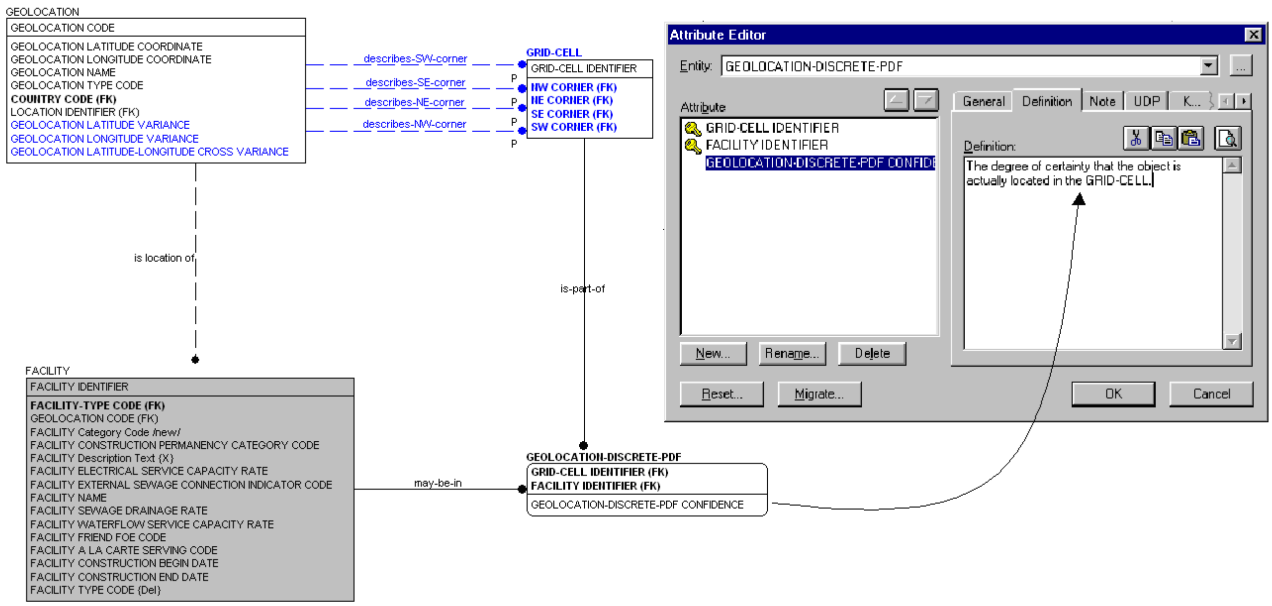


Figure 10. When the PDF Should Not Be Characterized Parametrically

of a new instance or an update to an existing one. If new, association hypotheses to prior measurements would be formulated, along with activations to transmitter type, and perhaps even transmitter instance hypotheses. These would then spread to the host equipments, platforms, and facilities, and then to the possible organizations, missions, and actions.

"Propagation" here is being used in the broad sense, from Bayesian, Dempster-Shafer, and neural-network, to state-estimation algorithms, i.e., any sort of method that would influence belief regarding the state(s) of adjoining nodes. The tasking model would retrieve the algorithm, invoke it to propagate hypotheses and compute states and uncertainties.

7 Computational Methods

This section addresses computation issues in executing the instrumented model so that it becomes an inference network. Some of the issues are applicable to spreading activation networks in general, are discussed extensively in the literature [2], [26, and are beyond the scope of this paper. The discussion that follows, instead, merely shows that it is possible to execute the afore-discussed model and that problems are no different than those encountered in inference networks in general.

7.1 Propagation Formulae as a Variation of Methods

The structure discussed in the preceding sections provides a flexible structure for storing various inference algorithms, in a manner analogous to methods in OO databases; in a relational DBMS, they would be "stored procedures". The propagation formulae would reside at a "type" level and would be applied to instances depending on the type-instance hypotheses. For example, if aircraft "tail number" A123 could be a MIG-29, then the type MIG-29 (or a higher superclass) would contain propagation formulae applicable to the instantiated, and perhaps new proposed, propagations from the instance A123. The propagation paths would be created from the set of path types specified in the superset of associations from the class object AIRCRAFT and its superclases.

7.2 Tasking Model

The tasking model would be a governed-automata model. The activations from any belief update (or new belief information) will generally spread in a non-ordered (asynchronous) manner. A task executive would control the automata through the parceling of computing resources, not necessarily the direction or manner of propagation. The methods or stored procedures at nodes would request tasking for any activated associated nodes, with a notation of the instances updated. DBMS "triggers' are an example of how automata could be stimulated to action. Governance is necessary because the computing demands of activation networks can be large so it is necessary to prioritize the activations in the most profitable manner. Many of the belief updates will be of low immediate consequence. The tasking request could include the degree of change of the input.

Unfortunately, the automata cannot estimate utility. It would be up to the task executive, perhaps with an associate, to discern if any sets of requested tasks seemed to be leading in productive directions using information utility measures along the lines in [27]. The object of interest, along with its properties and specific values, could be used by a separate goal monitor to inversely stimulate a background copy of the network, thereby spreading activation from desired information to evidence, in a kind of sensitivity analysis.

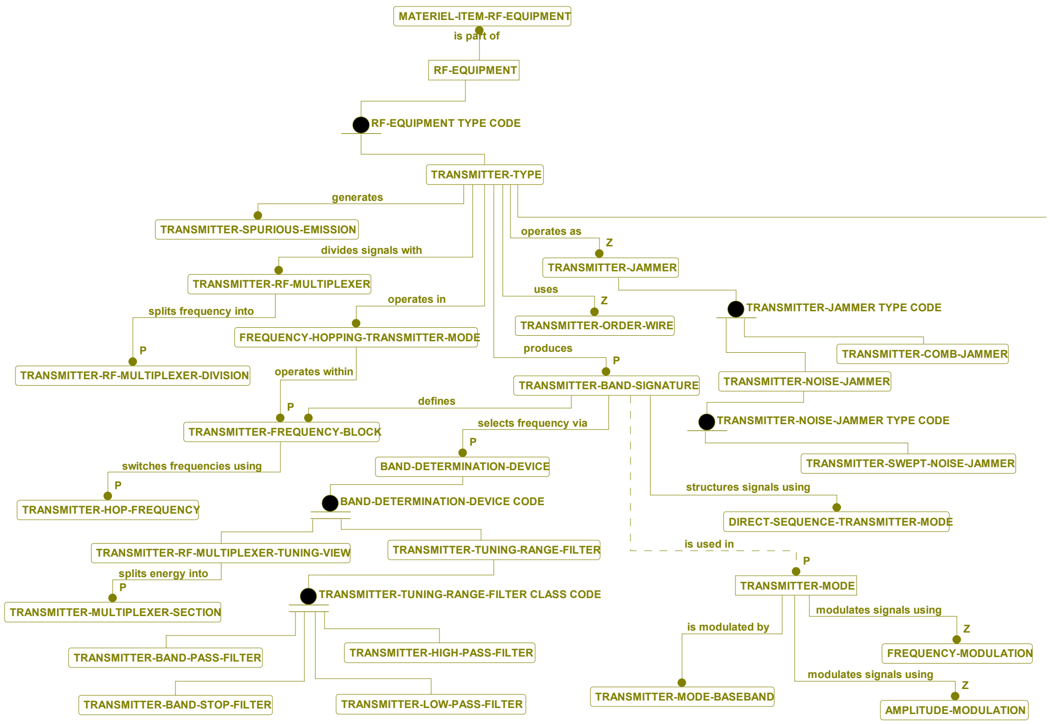


Figure 11. Sensor Measurements Related to Equipment

The third concern is knowing when the spreading is done. This is a common problem in many connectionist structures; see [28] for examples of the problem in neural networks. Examples of techniques to end the spreading are energy functions (e.g., Liapunov) and activation path thresholds. In the former, a cost function computes the energy of the spreaded activation and considers it to be complete if it reaches local minima or an established threshold. In the latter, a threshold is established (or computed dynamically) for each activation. If the activation value is too low, it does not fire. These generally accumulate at the node so that successive updates at the source node can build up energy that may then overcome the threshold.

A forth feature is reduction, if not elimination, of uncontrolled feedback to prior nodes. Directedness alone can alleviate some of this problem, but not all, as there are cases where two nodes can mutually influence each other. Proper structure of the inter-node influencing can also help; by making sure node influence is by truly causal factors. Unfortunately, this may run counter to independence architecture of the network. Cessation or de-prioritizaiton of insignificant activations, as part of the tasking and activation cessation controllers just described, also helps. Evidence tracking is insufficient, because a node’s updated state could be based upon multiple neighbor-node updates. Similarly, pedigree tracking is generally impractical because the pedigree could include all the node states, back to the start of the system that influenced current state. An approximation might be to track only significant and current pedigree. Maintaining with the each belief hypothesis, an accumulation of the inputs that most contribute to current belief could do this. A more complete treatment of this problem is beyond this scope of this paper.

7.3 Tools

The scale of fusion networks as described herein is large and the design will become intractable unless tools are applied. Certainly an E-R tool that automatically or near-automatically generates a database is important so that the network can evolve as new entities, attributes, and relationships are discovered in the real world. Using an off-the-shelf DBMS will also be helpful since it will have all the insert, update, relationship, and triggering mechanisms built-in. If the fusion centers operate in a distributed manner, many DBMS's will also have replication services that can automatically communicate node updates within the replica family. This may mean operating with massive amounts of solid-state memory and in an architecture that can fade no-longer-used values off to archive or deleted in a non-interfering manner. An example of specialized a memory-resident DBMS is TimesTen [29]. Lastly, it will be important to employ object-oriented techniques and, if possible, tools, so that the propagation algorithms, operating at the appropriate class level, are encapsulated as methods to apply to the instances of the class.

8 Conclusion

Powerful information modeling, database, and OO tools and techniques have evolved over the past 10 or 20 years that can provide a foundation for large-scale fusion systems. Because this foundation is relatively stable, being grounded in the fundamental semantics of the enterprise, and because it is transparent, many types of fusion algorithms can be unified within it and operate in an integrated manner over a

Copyright 2002 © Silver Bullet Solutions, Inc.

broad spectrum of information. Algorithms can be updated or changed in a modular manner, with little to no effect on the rest of the system. This enables a broad community of participation in the fusion system. Due to the broad scope of modeling that is possible and the ability to employ custom but modular algorithms, most suitable to the belief at hand, this approach will allow the development of very-large fusion systems.

9 References

[1] Waltz, E., Llinas, J., Multisensor Data Fusion, Artech House, Inc., Norwood, MA, 1990

[2] Pearl, J., *Probabilistic Reasoning in Intelligent Systems, Networks of Plausible Inference*, Morgan Kaufman Publishers, Palo Alto, CA, 1988

[3] Sowa, J.F., *Semantic Networks*, http://jfsowa.com/pubs/semnet.htm

[4] Hammer, H., McLeod, D., "The Semantic Data Model: A Modelling Mechanism for Data Base Applications", *Proceedings of the ACM SIGMOD International Conference on Management of Data,* 1978

[5] Date, C.J., *An Introduction to Database Systems*, Addison-Wesley Publishing Company, Inc., 1986

[6] http://ca.com/products/alm/erwin.htm

[7] Saake, G., Conrad, S., Schmitt, I., Turker, "Object-Oriented Database Design: What is the Difference with Relational Database Design?", in *Object World Frankfurt '95, Conference Notes*, Zicari, R., editor.

[8] http://www-datadmn.itsi.disa.mil/datadmn/

[9] Hammer, H., McLeod, D., "The Semantic Data Model: A Modelling Mechanism for Data Base Applications", *Proceedings of the ACM SIGMOD International Conference on Management of Data,* 1978

[10] Bachman, C., Daya, M., "The Role Concept in Data Models", *Proceedings of the 3rd International Conference on Very Large Data Bases,* IEEE, 1977.

[11] Falkenber, E., "Concept for Modelling Information", in Modelling in *Data Base Management Systems*, Nijssen, G.M. (ed.), North Holland Publishing Company, 1976

[12] Stamper, R., "Physical Object, Human Discourse, and Formal Systems", in *Architecture and Model in Data Base Management Systems*, Nijsse, G.M. (ed.), North Holland Publishing, 1977

[13] Chen, P., "The Entity-Relationship Model – Towards a Unified View of Data", ACM Transactions on Database Systems, 1(1), March 1976.

[14] Codd, E., "Extending the Database Relational Model to Capture More Meaning", *ACM Transactions on Database Systems*, 4(4), 1979

[15] *Federal Information Processing Standard (FIPS) 184*, General Services Administration, Washington, D.C., 1993

[16] Rumbaugh, J., Jacobson, I., and Booch, G., The *Unified Modeling Language Reference Manual*, AddisonWesley, 1999

[17] Sowa, J.F. (ed.), "Current Issues in Semantic Networks", in *Principles of Semantic Networks, Explorations in the Representation of Knowledge*, Margan Kaufmann Publishers, Inc., San Mateo, CA, 1991

[18] Sowa, J.F., *Knowledge Representation, Logical, Philosophical, and Computational Foundations*, Books/Cole, Pacific Grove, CA, 2000

[19] Cravo, M.R., Martins, J.P., "Path-Based Inference Revisited", in *Current Trends in SnePS – Semantic Network Processing System*, Springer-Verlag, Buffalo, NY, 1989

[20] McDaniel, D.M., *Electronic Warfare Identification*, *SBIR, Final Report,* Space and Naval Warfare Systems Command, San Diego, CA, 1996

[21] McDaniel, D.M., "An Information Fusion Framework for Data Integration", in *Proceedings of the 13th Software Technology Conference,* 2001

[22] Keithley, H., *Multi-INT Fusion Performance,* Joint C4ISR Decision Support Center, OASD(C3I), Washington, D.C., 2001

[23] Kumar, D. (ed.), *Current Trends in SnePS – Semantic Network Processing System*, Springer-Verlag, NY, 1990

[24] Ee-Peng Lim; Srivastava, J.; Shekhar, S., "An evidential reasoning approach to attribute value conflict resolution in database integration", *IEEE Transactiona on Knowledge and Data Engineering*, Oct. 1996, Volume: 8 Issue: 5

[25] Lee, S.K., "Imprecise and Uncertain Information in Databases: An Evidential Approach", in *Proceedings of the 8th International Conference on Data Engineering*, IEEE, 1992

[26] Liu, J., Maluf, D, Desmarais, M., "A New Uncertainty Measure for Belief Networks with Applications to Optimal Evidential Inferencing", *IEEE Transactions on Knowledge and Data Engineering,* May/June 2001, Vol. 13, No., 3, IEEE

[27] Keithley, H., "An Evaluation Methodology for Fusion Processes Based on Information Needs", in *Handbook of Mutisensor Data Fusion*, ed. By Hall, D. and Llinas, J., CRC Press, New York, 2001

[28] Simpson, P., *Artificial Neural Systems*; Pergamon Press: Oxford, 1990

[29] http://www.timesten.com

Copyright 2002 © Silver Bullet Solutions, Inc.