An Analysis of Existing Ontological Systems for Applications in Manufacturing and Healthcare

1.0 Introduction

The objective of the work described in this paper is to move closer to the ultimate goal of seamless system integration using the principle behind ontological engineering to unambiguously define domain-specific concepts. A major challenge facing industry today is the lack of interoperability between heterogeneous systems. This challenge is apparent in many sectors, including both healthcare and manufacturing. Current integration efforts are usually based solely on how information is represented (the syntax) without a description of what the information means (the semantics). With the growing complexity of information and the increasing need to completely and correctly exchange information among different systems, the need for precise and unambiguous capture of the meaning of concepts within a given system is becoming apparent.

The approach for this project is to analyze current ontological systems to determine which is most suitable to model the concepts in both the healthcare and manufacturing domains. Examples of ontological systems include the Unified Medical Language System (UMLS[1]), CYC[2,3], and the Ontolingua server. For the manufacturing-related ontologies, the project will move to formally identifying and modeling concepts (and definitions of those concepts) from various manufacturing domains (e.g., process specification, product modeling, resource representations, etc.) in this ontological system. At this point, an analysis can help to identify inconsistencies in the use of terms among various domains as well as help to establish a means to generalize these terms to a level that is common among the domains in question.

The output of the entire work documented in this paper will be a taxonomy of terms and concepts, formal definitions of exactly what each of those terms and concepts means, and how they interrelate. Although it would be impossible to create a complete taxonomy of every interpretation of every term, a high-level, extensible subset of this taxonomy will be created to serve as a basis for future, domain-specific additions and specializations. This shared understanding of concepts could be used to integrate applications and systems that function towards a common goal.

This paper documents the results of the first phase of this project – that of analyzing existing ontological systems to determine which is most appropriate for the manufacturing and healthcare domains. In particular, this phase involved the exploration of efforts that are studying both the uses of ontologies in the general sense and those that are using ontologies for domain-specific purposes.

2.0 Why Ontologies for Manufacturing?

This section considers what value the investigated ontologies might provide to the area of information technology within manufacturing. Communication and context are important notions for understanding the role of the investigated ontologies vis a vis other technologies. This and the related concepts of formality, ground, and context availability are discussed. Three areas of potential benefit are then considered: unambiguous communication, standards-making and semantic-alignment efforts, and future industrial information infrastructures.

[1] The Unified Medical Language System (UMLS) is a registered trademark of the U.S. National Library of Medicine.
[2] CYC is a registered trademark of Cycorp Inc.
[3] No approval or endorsement of any commercial product in this paper by the National Institute of Standards and technology is intended or implied.

2.1 COMMUNICATION, MEANING AND CONTEXT

In this paper, "communication" has the following meaning: One communicates to another with the expectation that at some time thereafter the receiver will produce a behavior that is in some way consistent with the initiator's intention. For communication to succeed, the initiator must have some understanding of the context under which the receiver is operating and the relationship between the message it designs and the behavior it desires from the receiver.

This definition of communication begs the question of what is "meaning." Along the lines of Bloomfield [BLO33] there is at the very least this sense of meaning to communication: it sets conditions for satisfaction. If I say "Pick up the book." and you do so, this is evidence of a relationship between a representation and a perceived reality. Philosophers might argue that there is more or less meaning in 'meaning', but computer scientists need not care; they are interested in the behavior of programs.

The phrase 'understanding of the context' in the above definition of communication is fraught with implication. Roughly, context is an environment, a "place" where things occur or an utterance is made. The anticipation of, and preparation for, a particular environment is a basic design question.

It is largely a matter of how context is established and used that differentiates various software design and development approaches. A complete study of this issue is well beyond the scope of this paper, but for the purpose of understanding the value of the investigated ontologies it is worthwhile to establish some dimensions on the term "context" and place the investigated ontologies and related technologies in these dimensions.

In order to cut the notion of context down to a manageable size, we first make the distinction among three fundamentally-different means by which software technology embodies context. Along the lines of Varala et al. [VAR91], these technologies are "enactive," "emergent," and "symbolic".

Neural nets could be viewed as representative of enactive technology. In neural nets, an associative memory embodies the relationship between demands and behaviors in response to those demands, that is, context is reflected in the trained associative memory. Enactive technology is outside the scope of this paper.

Genetic algorithms could be viewed as representative of emergent technology. In the emergent approach, context could be reflected in the statistical distribution of an individual design within a spectrum of possible designs. Each individual is an attempt to encode (and cope with) the context. Emergent technology is outside the scope of this paper.

Symbolic technology includes procedural (e.g. object-oriented), constraint, and logic programming, ontologies, and natural language. Unlike the others, symbolic technology is linked inextricably with language and hence meaning and context can be communicated among individuals, not just embodied by them. Roughly speaking, dictionaries, thesauri and computer-based ontology systems are each built upon networks of symbols related to each other by various notions of resemblance. In terms of such networks we can state two fundamental problems of communication as (1) individuals seldom share a common network of symbols and (2) establishing a context in a network is difficult.

Regarding the first of these problems, people do not communicate using identical networks. Our use of language is idiosyncratic and a result of life experience. In some sense the problem is worse for computers than it is for people. As Quine [QUI87] points out, among people "The coherence is no coincidence, for the network is self-corrective." Were I to use the word "hello" to mean "goodbye", someone would correct me and I would discontinue this erroneous practice. Such correction capabilities usually don't accompany software.

With respect to the second problem, most people can recall entering into the middle of a conversation and totally misinterpreting what was being discussed. In software communications, context is almost always assumed, with potentially disastrous consequences.

In computer technology, at least, it is somewhat artificial to say that these are two different problems. Although errant software may not have critics to re-align the network, the software development process does. In essence, all problems of context in software (excepting perhaps some artificial intelligence applications) are resolved by the software developers on both sides of an information exchange coming to agreement on what the information exchanged means (problem 1). The arrangement they choose might be ad hoc or by reference to a consensus standard, for example.

The software developers above might never know, nor care, whether they are developing interfaces based on identical understanding. What is essential is that the software behaves as expected. Much the same can be said of the interactions of people.

This discussion leads us to a few questions relevant to the positioning of the investigated ontologies vis a vis other technologies:

1. Need we do more than get the behavior right?
2. How do we know that we are in agreement well enough to get the behavior right?
3. What holds the ontology?

Of course, good answers to these questions do not come easily. We can not respond completely at this point in the investigation. Towards an answer to any of these, we consider the notions of ground, formality, seamlessness, and context availability.

2.1.1 GROUND

Webster's defines "ground" as "the foundation for an argument, belief or action." Ontologies are one technology where the notion of ground is prominent. In some sense, unambiguous communication of information is enabled by relating consensus domain terminology to widely held ground terms. There are however, overriding issues in making this goal a reality. This is discussed in the section on unambiguous communication.

Some ontology systems (for example CYC) provide a mechanism to allow reasoning under multiple sets of ground terms ("assumptions") [deK86]. Intuitively this appears useful in manufacturing situations but more consideration is necessary.

It is a mistake to assume that only ontological systems make explicit use of ground. For example, the Eiffel programming language uses "contracts", which are relations among communicating components whose interaction is based on precisely-defined specifications of mutual obligation to support ground for action. In [JEZ98], Jézéquel and Meyer argue that an object-oriented 'reuse error' caused the $500M crash of an Ariane rocket. They suggest that the crash could have been prevented had the notion of contracts between objects been used.

The contract paradigm, an "operationally-active" flavor of the notion of ground, enables reliable reuse in object-oriented programming. The contract paradigm complements the current thrust in standards for distributed, object-oriented systems such as those of the Object Management Group [OMG98].

2.1.2 CONTEXT AVAILABILITY IN INFORMATION EXCHANGE

In the design of communicating systems, one may choose to resolve all questions of the context of information exchanged at design time. Alternatively, the exchange might include context-setting information, or 'meta-level' information.

Many high-performance systems do not communicate much context information between each other; the context of the information exchanged was agreed upon by the system designers a priori and is implicit in the software. The exchange of a single property in STEP (STandard for the Exchange of Product model data) [ISO94] technology, on the other hand, includes several context-setting references: references to the measurement units used (e.g. newton/meters), the various representations of the property (e.g. as thermal energy, as mechanical energy), the technical discipline it concerns (e.g. “structural analysis”), and the product lifecycle (e.g. “design”). [DAN97].

Within the investigated ontologies, the meta-level is available as a reference that could lead to ground terms. What is not as clear at this point in our investigation is what manner meta-level information is made available in information exchange. We may study the Knowledge Interchange Format (KIF) in our second year effort as a means towards this end.

2.1.3 SEAMLESSNESS

A context or meta-level may be available but only in a different technology than the problem solving machinery. This is a common occurrence (e.g. the OMG implementation repository). Seamlessness refers to the continuity (or 'transparency') of the information content with its context-setting information. That is, in a seamless environment, the problem solving machinery can get context-setting information without appealing to another for service.

The principle question here, with respect to the investigated ontologies, is how to marry the technology to the 'problem solvers'. That is, how should ontologies be interfaced to applications such as schedulers and workflow systems? We did not attempt to answer this question in this year's investigation.

2.1.4 FORMALITY

At times, formality in ontologies seems to mean the degree to which the ontology resembles mathematical logic. Resemblance to mathematical logic in itself however does not suggest a purpose for formality. We suggest the following:

- Formality is about making valid inferences (and thus getting expected behaviors).

- Formality is about traceability to ground.

- Formality is about enabling computational tools (to manage complexity etc.).

There are many notions exchanged among engineering and manufacturing systems for which it might be quite inefficient to attempt to represent the ideas in anything but traditional mathematical terms. Computer-Aided Design (CAD) geometry is of this sort.

2.2 ANSWERS TO 'WHY ONTOLOGIES?'

The previous section provides a foundation of ideas concerning communicating systems. In this investigation, we have not yet resolved many design questions regarding how the use of ontologies may be employed. Assuming that there are reasonable answers to these questions, however, we may still ask what added value the investigated ontologies might provide to the development of a manufacturing information infrastructure. Here we consider the contribution ontologies might make to eliminate ambiguity in communication, and in the standards making process.

2.2.1 UNAMBIGUOUS COMMUNICATION

We consider two questions with respect to the investigated ontologies and the goal of unambiguous communication:

1. Where does the problem of ambiguity reside?
2. Do communicating systems need access to ground terms?

In regards to the first question, we note that, in the opinion of the authors, manufacturing systems have lagged financial systems in their ability to exchange data. One explanation for this might be that less time and effort has been invested in the integration of manufacturing systems. There are other equally valid explanations:

- Manufacturing data and its interrelationships are complex, perhaps much more so than financial data.

- There is no universally accepted meaning to terms used in manufacturing.

It is important to note that these two points are statements about the nature of manufacturing information itself: their solution to the problems raised in these points requires knowledge of manufacturing foremost. Information technology isn't the issue here. The problem is getting individuals in whole industries to agree on the meaning of perhaps thousands of terms such as "part version" and "part revision." This is about raising the biblical “Tower of Babel” and putting in its place a consistent terminology. It requires a large effort, but one that is (and has been) essential to the goal of unambiguous communication. Standards efforts such as IEC 61360-4, a dictionary of standard terminology for electronic components [ISO97] are representative of the sort of work that must be done.

In summary to the first (and most crucial) question, we recognize that our answer leads back to the standards making process, and in effect, the choice of modeling concepts with formal semantics versus more conventional information modeling technology for the purpose of information exchange appears inconsequential over the short term (15 years?). The development of domain consensus terminology is most important presently.

With regards to the second question, (whether communicating systems need access to ground term) we consider an alternative strategy, that of the STEP standards.

In STEP, manufacturing notions are related to more fundamental notions through an interpretation process. This interpretation process has similarities with Schank's conceptual dependencies [SCH81]. In STEP, a collection of attributed entities (integrated resources) possessing partially-defined semantics (i.e. under-defined building block enabling wide applicability) are made to represent specific notions through their aggregation, interrelation, specification of values, and subtyping. Entities in a network identify the context (e.g. engineering and industrial discipline) and measurement units. In STEP, it is, in fact, this circumscribed collection of interrelated entities that represents the property, not an individual entity that possesses reference down to ground terms.

The distinction with respect to 'grounded' ontologies here is that within the STEP standards, meaning is fully described, provided that a property identifies its context, measurement units, and property name *and* this property name designates, without ambiguity, a notion that is the product of industry consensus on the terminology of that property.

In practice, however, the use of STEP does not always guarantee this consensus of property terminology.

The difference, then, between STEP and the investigated ontologies is the point at which each goes outside it’s domain to establish meaning. STEP is “flat”, beginning and terminating in specific industrial domain terminology; therefore, not going outside itself to establish meaning. The investigated ontologies are “bottom-up”, beginning with some fundamental notions and building towards more specific notions, which to date, have not encompassed many notions of importance to industry.

Communicating systems rarely need access to ground terms. The exception to this is those few systems that mediate data. The problem of bringing meaning to the exchanged data always leads back to reference to shared understanding. For this reason we conclude that ontologies do not offer significant benefit towards making current information exchange methods more reliable.

2.2.2 SHARED TERMINOLOGY AND SEMANTIC ALIGNMENT

As suggested above, there is a need for comprehensible, industrial terminology developed through consensus. As increasing amounts of the work involved with bringing a product to market are possible on the Internet and as increasing amounts of the supply chain become integrated, the audience for this terminology widens. A terminology shared among manufacturers of centrifugal pumps is most useful when it can also be understood by makers of bearings for these pumps and builders of process plants using the pumps.

Of course, various usages of the same term have grown up in various industries (e.g. "Stator" means one thing to a manufacturer of turbine engines and another to a manufacturer of automobile alternators). Because the investigated ontologies may be founded on notions common to both usages, it is possible to recognize when usages differ.

Although this aspect of the studied ontologies has not been a central focus of this year's project efforts, we expect it to be an area where the investigated ontologies may provide great benefit.

With regards to the question of who will write ontologies, it is our guess that some domain experts may be willing and able to become conversant in the language of an ontological system provided that they can see a clear benefit. Some manufacturing domain experts have tackled the equally challenging task of learning STEP technology and the EXPRESS language.

2.2.3 INDUSTRIAL INFORMATION INFRASTRUCTURE

Distributed objects, agents, integrated workflow, and supply chains are common directions emerging in the development of an industrial information infrastructure. This mode of operation implied by these terms emphasizes ad hoc access to objects. This is in contrast to the more traditional approach of organizing systems around the semantics of a shared database. This suggests that ad hoc access to shared meta-data and terminology might also prove useful in future information systems. If this turns out to be true, then technology such as the investigated ontologies may prove to be essential in supplying a terminology and meta-data in computational form.

2.3 SUMMARY

Assessing the value of ontologies (or any other technology) to the problem of communicating manufacturing information is, in part, a matter of determining whether it is the simplest possible means to establish a context on exchanged data under which valid inferences (and thus expected behaviors) can be achieved.

The investigated ontologies provide seamless access to its meta-data and ground terms in a computational, formal form. However, one cannot say absolutely whether seamless systems with access to meta-data and possessing a theory of ground are better or worse designs because they possess these attributes. Many very successful, high-performance systems will continue to possess seams, no theory of ground, and no access to meta-level information. The genius of system design is in part in making the right choice with respect to how context is established and used in communication.

The investigated ontologies can contribute significantly to the alignment of consensus domain terminology.

3.0 Approach and Major Findings for Manufacturing Analysis

A systematic approach was taken throughout this project to ensure that a proper cross-section of manufacturing-related, ontological systems were chosen, appropriate analysis criteria were determined, and a proper analysis was performed. The project started by doing a literature survey to determine what appropriate ontological systems were available. This survey included a thorough search of the web and numerous interactions with colleagues in the ontology field. From this survey, the following ontological systems were identified:

- ANSI Ad Hoc Group on Ontology Standards Representation [ANSI98]
- CYC [CYC98]
- Enterprise Ontology [ENT98]
- LOOM [LOOM98]
- MikroKosmos [MIKRO95]
- Ontolingua [ONTO96]

- Sensus [SENS98]
- SPAR (Shared Planning and Activity Representation) [SPAR98]
- STEP (Standard for the Exchange of Product model data) [ISO94]
- TOVE (Toronto Virtual Enterprise) [TOVE98]
- Wordnet [WORD98]

We then performed a high-level analysis of each of the above ontological systems and eliminated a few due to their lack of appropriateness to this project. In general, the project analyzed these ontologies against the following three criteria:

- the ontology's ability to represent manufacturing information (e.g., time-varying concepts, flow of materials, constraints, etc.),
- the amount of manufacturing information that was already represented in the ontology,
- the ability for the ontology to inference over the information represented.

The following systems were excluded from the analysis, along with the respective reason:

- ANSI Ad Hoc Committee on Ontology Standards – at the time the analysis was performed, this ontology was not mature enough to analyze properly (the concepts what were identified were "upper-level" concepts which were too generic for this work). In addition, since the upper level of CYC was to be merged with this ontology, an analysis this ontology would, in some ways, be redundant.
- Sensus – only a taxonomy of terms without definitions were provided and the concepts represented in this system had already been merged with CYC through the Ad Hoc Group on Ontology Standards work
- SPAR – at the time this analysis was performed, SPAR was very high-level and not mature enough to analyze.
- STEP – it was too limited in domain (only product data), there were no formal definitions of concepts, and from the project participants' previous work with STEP, we know it would not be appropriate.
- Wordnet – it is more of an on-line super-dictionary than a knowledge base.

Once the ontological systems to be analyzed were determined, we moved on to determining the appropriate analysis criteria. There were two approaches suggested:

1. Base our analysis on a matrix on which various manufacturing-related fields would be along the y-axis (e.g. process planning, scheduling, machine capabilities, etc.) and general areas of interest would be along the x-axis (temporal issues, model issues, and material existence issues). In each cell, there would be an appropriate question that would be posed to the ontological system pertaining to its location along the x- and y-axis and the analysis would involve gauging how well the system could answer that question.
2. Base our analysis on typical manufacturing scenarios. This would involve the identification of appropriate manufacturing scenarios, extraction of concepts inherent to that scenario, grouping of concepts into appropriate categories, and development of inferencing questions that are based on those concepts. We would then see how well existing ontological systems could model those concepts and determine how well they could answer questions pertaining to those concepts.

The project decided to go with the second approach because it seemed to provide a more complete and accurate analysis than the first. The CAMILE "Factory from Hell" scenario [CAMI91] was identified as an appropriate scenario for our manufacturing analysis. This scenario was developed by Ken McKay as part of an assignment through CAM-I (Consortium for Advanced Manufacturing, International) . The scenario details a fictitious factory (based heavily on knowledge gained through site visits to actual factories) including information on many departments and the decision-making processes that occur throughout the development of a product. The concepts, which were detailed in the scenario, were extracted and grouped into manufacturing-related categories. The chosen categories were (in no particular order):

a) Penalties
b) Costs

c) Financials
d) Scheduling
e) Process Planning
f) Product Configuration
g) Resource Planning
h) Resources
i) Inventory
j) Batches/Lots
k) Orders
l) Customer/Vendor
m) Scrap/Rework
n) Manufacturing Execution

Using these categories and the concepts identified in each category, we initially examined each of the ontological systems to determine how well they could represent those concepts. Namely, we rated each ontology with respect to the following four categories:

1. Required concepts are not represented in ontology. Related information infrastructure is not available and must be modeled before concepts can be represented.
2. Required concepts are not represented in ontology. Related infrastructural concepts are available. Modeling of required concepts could take place primarily by combination of existing concepts.
3. Representation of required concepts could be achieved through specialization or minor modification of existing concepts.
4. Required concepts are available in ontology and would require either trivial modifications or none at all.

During the initial phases of this analysis, we found that a few other ontologies were not appropriate for further analysis for the reasons described below.

- LOOM – it is a language and environment. It is not an ontology itself but is quite suitable for implementation of projects using ontologies. Therefore, LOOM would not be appropriate for the development and modeling of a manufacturing ontology.
- MikroKOSMOS – its purpose is to provide a general mechanism for mapping meaning between languages. As such, it has been developed with different capabilities and design structure than would be needed for a manufacturing ontology. Specifically, Mikrokosmos provides no inferencing capability to answer questions that are not explicitly answered in the knowledge base.
- Ontolingua – it is an ontology-authoring tool and not an ontology itself. Since this body of work is a development environment, it is not appropriate to attempt to evaluate its direct applicability to manufacturing.

For the above reasons, these ontologies were not further analyzed. The remaining three ontologies, CYC, Enterprise Ontology, and TOVE, were then analyzed in further detail. The results of this analysis can be found in Section 5.0.

The next section provides an overview of the analyzed ontological systems.

4.0 Manufacturing Ontological Systems Investigated

The following table summarized the major points related to the ontologies that were investigated. A more detailed description of each of these ontologies can be found in the subsections below.

Ontology	Domain	Purpose	Provides Inferencing?	Development Framework or Full Ontology
CYC	Generic	Enable common sense reasoning about the world	Yes	Full ontology
Enterprise Ontology	Business enterprise and organization modeling	Comprehensive ontology whose main groupings consist of activities, organization, strategy, marketing, and time.	No	Full Ontology
LOOM	Generic	A language and environment for constructing intelligent applications	Yes (forward, truth maintenance)	Development Framework
Mikro-KOSMOS	Knowledge-based translation of natural language	Translate natural language text from one language to another via a language-neutral text meaning representation	No	Full Ontology
Ontolingua	Generic	Development environment and authoring tool for the creation of modular, reusable ontologies.	No	Development Framework
TOVE	Enterprise integration	Provide a generic, reusable data model including shared terminology and meaning that each agent can jointly understand and use	Yes	Full Ontology

Table 1: Summary of Ontologies Investigated

4.1 ANSI Ad Hoc Committee on Ontology Standards

The goal of the ANSI Ad Hoc Group [ANSI98] (associated with the ANSI X3T2 committee on Ontology Standards) is to merge the upper level ontologies of many of the well-known ontological systems (CYC, Pangloss, Penman, Wordnet, etc.). An "upper level ontology" is an ontology of the most general conceptual categories. There are a number of such ontologies out in the world that have proved very useful in natural language processing and other AI oriented applications, as well as in enterprise modeling and database integration. The challenge is that it is difficult to translate between these applications because of the differences in their upper level ontologies. The purpose of the standard will be to provide a sort of ontological baseline to support translation and integration between ontology-based applications, and hopefully to serve as the starting point for future upper level ontologies.

Included in this group Pat Hayes (Univ. of Western Florida), John McCarthy (Stanford University), John Sowa, Fritz Lehmann (Cycorp Inc.), Ed Hovy (University of South Carolina), and Peter Simons (University of Leeds and Ontek, Inc.).

At the time the analysis was performed, all that was available from this group was a high level taxonomy of terms without any definition of what the terms meant. We assumed that the location of any term within the taxonomy was meant to serve as a loose definition of the term. However, because this ontology standard was being adopted by other systems that we were analyzing, such as CYC, the analysis of those other systems would allow us to analyze the ontology standard indirectly. In addition, because those systems provided additional capabilities that the ontology standard alone did not (e.g., inferencing capabilities, formal definitions of terms, user interfaces, etc.), the respective systems would be a more appropriate choice for use to model a manufacturing ontology. For these reasons, this Ontology Standard was not investigated any further.

4.2 CYC

CYC [CYC98] is a very large, multi-contextual knowledge base and inference engine developed by Cycorp. The goal of the CYC project is to construct a foundation of basic "common sense" knowledge--a semantic substrate of terms, rules, and relations--that will enable a variety of knowledge-intensive products and services. CYC is intended to provide a "deep" layer of understanding that can be used by other programs to make them more flexible.

A drawback to CYC is that its level of knowledge is so "deep" as to be unintuitive to all but CYC knowledge experts. Higher-level knowledge is left to application developers. Not surprisingly, there are large gaps in CYC 's higher-level KB as it has been extended only to support whatever application was required for its use. Only some aspects of these extensions are publicly available. Manufacturing is not well represented by the KB. Parts of the healthcare area are well covered; however, large gaps exist.

The CYC technology is composed of the knowledge base and inference engine, the CycL representation language, interface tools and application modules. Cycorp is currently working on tools to ease the difficulty of adding to the KB.

The CYC KB is divided into "microtheories", each of which is essentially a bundle of assertions that share a common set of assumptions. Typically, microtheories are focused on a particular domain of knowledge, a particular level of detail, a particular interval in time, etc. The microtheory mechanism allows CYC to independently maintain assertions that are contradictory, and enhances the performance of the CYC system by focusing the inferencing process. At the present time, the CYC KB contains tens of thousands of terms and several dozen hand-entered assertions involving each term.

CycL, the CYC representation language, is a large and flexible knowledge representation language. It is essentially an augmentation of first-order predicate calculus (FOPC), with extensions to handle equality, default reasoning, and some second-order features. (For example, quantification over predicates is allowed in some circumstances, and complete assertions can appear as intentional components of other assertions.) CycL uses a form of circumscription, includes the unique name assumption, and can make use of the closed world assumption where appropriate.

CYC 's inferencing capabilities are discussed in the Section 5.2.1.

4.3 Enterprise Ontology

The Enterprise Ontology [ENT98] was built as part of the large Enterprise Project at the Artificial Intelligence Applications Institute at the University of Edinburgh, in collaboration with industry partners. The focus of the project is to promote the use of knowledge-based systems in enterprise modeling and organizational support. The result of this initiative was an Enterprise Toolset, one component of which is the Enterprise Ontology.

The Enterprise Ontology is an in-depth ontology, including terminology as well as definitions of and relationships among terms, for support of business enterprise and organizational modeling. The purpose of the ontology is to provide a formal model of an enterprise on which an enterprise modeling framework, methods and tools could be built. The motivation for a formal model is to establish an agreed-upon, shared understanding of the content and characteristics of an enterprise.

The Enterprise Ontology includes over 90 different concept classes and over 60 relations between concepts. The content of the ontology was created originally using natural language, and was then modeled formally in Ontolingua [GRU93] using the Ontolingua ontology-development environment [FAR96] described previously.

In order to represent concepts within the Enterprise Ontology itself, a *meta ontology* was developed, which includes more general modeling terms such as entities, relationships, roles, attributes, and so on. Building on these terms, the concepts in the Enterprise ontology are divided into five categories: activities, organization, strategy, marketing, and time. Activities are used to capture the concept of actions and doing, and related concepts such as the "doer" of an activity, resources used or produced, and so on. Organization concepts include entities such as people, machines, or corporations, and are used to model an organization's management structure. Strategy concepts involve the idea of planning and purposes. Marketing involves all sorts of concepts relating to sales, prices, vendors, customers, competitors, and so on. Time is used to represent timelines, specific points in time, or intervals of time duration. Of course, there are interactions among the various categories of concepts. For example an activity may take place over an interval of time as part of a plan.

The intent of the Enterprise Ontology is not to model specific types of enterprises, but to provide a general model that is oriented more towards businesses and organizations. Virtually all concepts and terms that are specific to manufacturing enterprises are missing from this enterprise model. However, the Enterprise Ontology is still viewed as a valuable resource because of the infrastructure it provides. The *meta ontology* provides a flexible set of primitives for building concepts, and since manufacturing enterprises are a subset of business enterprises in general, many of those aspects of a manufacturing enterprise that are not manufacturing-specific are present in the existing ontology. For instance, concepts such as resources, people, machines, and plans will have direct applicability within a manufacturing enterprise model. It should be noted that that in most cases, for application to a manufacturing enterprise, further specification of concepts existing within the current ontology will be necessary.

4.4 LOOM

"Loom is a language and environment for constructing intelligent applications. The heart of Loom is a knowledge-representation system that is used to provide deductive support for the declarative portion of the Loom language. Declarative knowledge in Loom consists of definitions, rules, facts, and default rules. A deductive engine called a classifier utilizes forward-chaining, semantic unification and object-oriented truth maintenance technologies in order to compile the declarative knowledge into a network designed to efficiently support on-line deductive query processing." [LOOM98]

As this quote makes clear, Loom is both a language and an environment. It is not an ontology itself but is quite suitable for implementation of projects using ontologies. Loom is written in Common Lisp and the Common Lisp Object System (CLOS) and is integrated easily into Common Lisp programs. The importance of Loom in this study is that it exemplifies the sort of infrastructure that exists to enable development of high-quality knowledge-based systems. Because Loom is not a commercial product (it is the intellectual property of the University of Southern California) there are fewer barriers to its use.

Our exploratory work with Loom suggests that it is easy to use. Although there may be concerns among some about Common Lisp not being a mainstream programming language, the development of a robust Common Lisp-based HTTP server has eased this problem somewhat.

4.5 MikroKOSMOS

The ultimate objective of the Mikrokosmos [MIKRO95] research project is to define a methodology for representing the meaning of text in a language-neutral format called a text meaning representation (TMR). This would provide a mechanism for knowledge-based machine translation (KBMT) of natural language text from one language to another (via an intermediary translation into a TMR). In pursuit of this goal, researchers at New Mexico State University have conducted a comprehensive study of linguistic and language-use phenomena. These phenomena have been encapsulated in various "microtheories", which are united through the control architecture of the KBMT system.

These microtheories assist the KBMT in translating to and from the language-neutral TMR. The TMR is obtained from analysis of the lexical, syntactic, semantic, and pragmatic information in the input text. This information is represented using elements of the MikroKOSMOS ontology. It is important to note that the TMR is also syntax neutral, since different languages have different syntax. As such, the TMR must avoid using (potentially) language-specific terminology, such as tense or clause. The TMR can also include stylistic factors, discourse relations, and speaker attitudes that may be present in natural language text.

The principle objective of the Mikrokosmos project is not directed at arbitrary queries of a specific knowledge base, but rather, a general mechanism for mapping meaning among languages. As such, it has been developed with different capabilities and design structure than would be needed for a manufacturing ontology. Specifically, Mikrokosmos provides no inferencing capability for answering questions that are not explicitly answered in the knowledge base. This capability is vital for providing useful information in a Manufacturing context. The Mikrokosmos ontology contains a wide variety of basic concepts related to manufacturing (e.g., drill, cut, and make), but it has very few detailed concepts that would be helpful for manufacturing. As such, implementing a manufacturing ontology using Mikrokosmos would require the development of tools for inferencing capabilities and general querying of the knowledge base, as well as adding a tremendous number of detailed concepts to the knowledge base.

4.6 Ontolingua

The Ontolingua [ONTO96] ontology development environment, developed at the Stanford University Knowledge Systems Laboratory, consists of a suite of authoring tools for creating and browsing modular, reusable ontologies. The set of tools provides a World Wide Web-based interface for ontology creation, allowing remote ontology creation or browsing of existing ontologies, many of which are available through the server Ontolingua Server at Stanford University.

The Ontolingua ontology development environment provides an extensive framework to support ontology authoring. Ontologies that are developed are modular and reusable, meaning that an ontology can "include" other ontologies and make use of their content simply by including references to them. Thus, when constructing a new ontology, many of the basic kinds of entities and infrastructure are already available by virtue of having been previously created by somebody else. These range from generic ontologies concerning algebra, time, and standard units, to more specific ontologies such as parametric constraints and mechanical components. Leveraging previous development efforts can result in significant timesaving for new ontology creation.

The Ontolingua ontology development environment models information using the Ontolingua language [GRU93], a language based on the Knowledge Interchange Format (KIF) [GEN92]. Ontolingua expands the basic first-order predicate logic formalism provided by KIF to also include syntax for an object-oriented representation (classes, instances, slots, relations, etc.) In addition to the web-based authoring interfaces, the development environment also provides translation into other knowledge representation languages, including Loom [MAC91], Epikit [GEN90], Generic-Frame [CHA97] and pure KIF.

The purpose of this paper is to evaluate ontologies and not ontology authoring tools. Since this body of work is a development environment, it is not appropriate to attempt to evaluate its direct applicability to manufacturing. However, because of its advantages (ease of use, availability of existing modular ontologies to leverage from, ties to KIF and translator facilities to interface with other knowledge representation languages), this environment would be a strong candidate for consideration if a new manufacturing-related ontology were to be built from scratch. Indeed, this development environment was used to model the Enterprise Ontology, which is one of the ontologies evaluated in this paper.

4.7 TOVE (TOronto Virtual Enterprise)

In order to support enterprise integration, it is necessary that a shareable representation of knowledge be available that minimizes ambiguity and maximizes understanding and precision in communication. Secondly, the creation of such a representation should eliminate much of the programming required to answer "simple" common sense questions about the enterprise. The goal of the TOVE [TOVE98] project is to create a generic, reusable ontology that has the following characteristics:

- provides a shared terminology for the enterprise that each party can jointly understand and use,
- defines the meaning of each term (a.k.a. semantics) in a precise and as unambiguous manner as possible,
- implements the semantics in a set of axioms that will enable TOVE to deduce automatically the answer to many "common sense" questions about the enterprise, and
- defines a symbology for depicting a term or the concept constructed thereof in a graphical context.

The TOVE reusable representation represents a significant ontological engineering of industrial concepts. All axioms and definition are specified natively in the Knowledge Interchange Format (KIF) [GEN92]. It also has presentations using the Frame Ontology from the Knowledge Systems Laboratory (KSL) (http://www.ksl.stanford.edu/) from Stanford and will shortly have a presentation in XML (eXtensible Markup Language) [XML98].

The work began by translating the ontologies developed at Carnegie Mellon from LISP into a C++ environment. The ontology was then modified and extended. Currently, the ontology spans: activities, state, causality, time, resources, inventory, order requirements, and parts. There has also been work to axiomatize the definitions for portions of our knowledge of activity, state, time, and resources. The axioms are implemented in Prolog and provide for common-sense question answering via deductive query processing. Future work focuses on the development of ontologies and axioms for quality, activity-based costing, and organization structures.

5.0 Discussion of Manufacturing Analysis

The results of this analysis showed that all three packages which were analyzed in detail (Cyc, Enterprise Ontology, and TOVE) were approximately equally in their ability to represent manufacturing information based on the information which already existed and the ability to specialize this information to make it appropriate for the manufacturing field.. However, the inferencing capabilities in CYC proved to be a bit more mature than the other two packages analyzed (see Section 5.2). Also, the close relationship that NIST and the ATP Ontology project have with Cycorp would allow the project to more easily leverage Cycorp staff's expertise while modeling the manufacturing ontology. For these reasons, the project decided to proceed with CYC to model the manufacturing ontology.

The one main concern was the fact that we would use a proprietary software package to model information that is meant to be in the public domain. This concern was alleviated because of the following:

1. The publicly-available "upper ontology" of CYC is currently exactly the same as the ANSI Ad Hoc Group for Ontology Standards' ontology (which can be represented in KIF or the CycL language). Because there is a bi-directional mapping and translators from CycL to KIF, if necessary, any work modeled in CycL can be easily converted to KIF. The CycL language is currently publicly available.
2. Through an agreement with Cycorp, any concepts that are needed to model manufacturing information in CYC, but which are not currently in the public domain, will be made publicly available.
3. There is currently work-in-progress in academia to develop an inferencing engine that will work on the CycL language. If this work continues to completion, there should be no concerns on having to use Cycorp's proprietary inference engine to be able to inference over the information in the manufacturing ontology.

Apart from these major findings ns, there were a couple of other interesting findings, discussed below.

5.1 Context in Ontologies

The main objective of ontology development is to develop a standard vocabulary or to predefine terminology in order to facilitate exchange of information. Ontologies help create a uniform basis for information exchange by enabling the representation and communication of the meaning of a given term. However, a secondary issue that must be addressed arises when a term has multiple valid definitions. Being able to represent these definitions formally does not solve the problem of knowing which definition to use in a given circumstance.

This problem is being addressed in several ontology development efforts through the use of contexts in ontologies (e.g. [CYC98], [MIKRO95], and [TOVE98]). Context, also referred to in some efforts as microtheories, allows additional information beyond specific, formal term definitions to be incorporated into an ontology. This contextual information may be represented implicitly or explicitly within an ontology. In the case of the Ontolingua Ontology Development Environment, modular ontologies are created and combined or included as components of larger ontologies. In one sense, this can be thought of as an implicit representation of context, since a term may be defined one way in one ontology and differently in another. MikroKosmos uses context to help resolve the meaning of words that could have multiple meanings. Although the way they do this is vague, it partially involves grammatical rule (e.g., adjectives follow nouns in Spanish, adjectives precede nouns in English). The placement of these words in a sentence provides the context to help to define what the words mean. CYC represents context using "microtheories", each of which is essentially a bundle of assertions that share a common set of assumptions, typically, microtheories are focused on a particular domain of knowledge, a particular level of detail, a particular interval in time, etc.

5.2 Inferencing in Ontologies

Inferencing, in general terms, is the ability for a system to deduce new information, which is not explicitly represented, from concepts that are currently represented in a knowledge base. For example, let’s assume that a particular manufacturing process (Process B) must be performed within 24 hours of the completion of another manufacturing process (Process A). In order for a scheduling program to decide when to schedule Process A, it must have access to certain information. Some of this information would be explicitly represented, such as the expected durations of Process A and Process B, the current time, and the standard hours that the factory is open. However, some of the information necessary may not be explicitly represented, such as, whether or not the factory is open “tomorrow”. This type of information would need to be deduced from information that is explicitly represented, such as, today’s date, today’s day of the week, scheduling holidays, and factory hours. An inference engine could provide this deductive capability to determine the information needed but not explicitly represented.

In the ontologies investigated, the tools were designed to work with specific representations; for example: 1) inference engines developed by Cycorp Inc. to work on their CycL representation, 2) a deductive engine developed with LOOM to specifically work on the LOOM knowledge representation, and 3) a set of tools developed all around the world to work on information represented in the Knowledge Interchange Format (KIF). The LOOM deductive engine is discussed in Section 4.4. The other two inferencing approaches are discussed below.

5.2.1 Inferencing in CYC

The CYC inference engine utilizes a variety of general, logical-deduction techniques, including modus ponens, modus tolens, and universal and existential quantification, as well as some specialized AI inferencing mechanisms, such as inheritance.

Because the CYC Knowledge Base (KB) contains hundreds of thousands of assertions, many inferencing mechanisms intended for smaller KBs are too inefficient. As a result, the CYC team has been forced to develop other techniques. CYC performs best-first search over proof-space using a set of proprietary heuristics, and uses microtheories to optimize inferencing by restricting search domains. CYC also includes several special-purpose inferencing modules for handling a few specific classes of inference. One such module handles reasoning concerning collection membership/disjointness. Others handle equality reasoning, temporal reasoning, and mathematical reasoning.

5.2.2 Inferencing Using KIF

Although numerous colleagues have noted that there currently exist inferencing engines for KIF, one were able to produce a single one. Even a web search did not uncover an engine's existence. True, a language called Prolog exists which provides inferencing capabilities for its own representation language, which is a restricted form of KIF, but nothing was found that uses KIF directly. Therefore, we can only conclude that if an inference engine does exist, it is certainly not easy to find.

6.0. Healthcare Ontologies / Unified Medical Language System

6.1. Overview

6.1.1. What is an ontology from a healthcare perspective

Similar to manufacturing, there is a pressing problem in the healthcare field that ontologies could help solve, namely, integration of often isolated healthcare-related representations. In this context an ontology can be viewed in two slightly different ways:

1. As a formal representation of pieces of information (e.g. ontologies for disease classification, nursing diagnosis, drug codes, etc.) that one would need to capture. This is the most common use of the word "ontology" in the healthcare arena (this is similar to the way the term is used within manufacturing in the previous sections) and will be used to mean this definition throughout the rest of the paper.
2. As a framework to help bring together pieces of information into a comprehensive system to provide a central location for all necessary information (e.g., similar to an architecture for an enterprise model in the manufacturing arena).

Ontologies exist in both of these areas, although, not surprisingly, there are many more ontologies that exist for the former than the latter. The following two sections describe the problems and the solutions in more detail.

6.1.2. Problems that healthcare ontologies are trying to solve

Similar to the manufacturing domain, there are multiple, isolated information models that represent different, but not necessarily orthogonal, subsets of the information needed and used in the healthcare field. Examples of these information models, hereby referred to as healthcare ontologies, are presented and described in Section 6.2. Currently, each of these models does very well at capturing the aspects of the healthcare field that they were created to represent, but they usually do not do well at sharing information interoperate with each other. This problem has hindered the possibility of bringing these ontologies together to create an "enterprise model" for healthcare – one that incorporates all aspects of healthcare information into a single framework in which information is easily shared and exchanged among various functions that use that information.

6.1.3. Approaches to solving the problem

There seem to be two distinct approaches to solving the integration problem described above. The first deals with the development of a single model that could facilitate either exchange or operations. In the case of exchange, this model would serve as an interlingua (a neutral representation) to tie together different representations that generally represent the same types of information. This would allow different companies that represent their information in different proprietary formats to be able to share information by first converting Company A's proprietary representation into the neutral representation and then converting the neutral representation into Company B's proprietary representation. An example of this type of representation can be found in LOINC [LOINC98] work described in Section 5.2.8.

In the case of an operational model, the model would serve as a SINGLE, complete representation of ALL information needed in the healthcare field. By creating a single representation, integration of multiple, smaller representations becomes a non-issue – all of the information one needs already exists as a single source. The disadvantage of this approach is that the benefit of previous work on modeling various aspects of the healthcare field becomes useless except for providing ideas on how to create the larger healthcare representation. An example of this type of approach is the Read Codes [Read98] work described in Section 6.2.13.

The second approach focuses on developing the framework, or the 'glue', to allow any existing or future representation to easily be integrated with each other. In this approach, there will most likely be an all-encompassing structure with formal specification of what concepts mean and how they all fit together which would allow a clear specification of the role that each individual representation could play in the overall structure. An example of this approach is the Unified Medical Language System (UMLS) [UMLS98], which is described in further detail in section 6.3.

6.2. Related Healthcare Ontological Systems

6.2.1. Arden Syntax

The Arden Syntax for Medical Logic Modules (MLMs) [ARDEN92] is a language for encoding medical knowledge. Each MLM contains sufficient logic to make a single medical decision. MLMs have been used to generate clinical alerts, interpretations, diagnoses, screening for clinical research, quality assurance functions, and administrative support.

With an appropriate computer program (known as an event monitor), MLMs run automatically, generating advice where and when they are needed. For example, one MLM warns physicians when a patient develops new or worsening kidney failure. Strictly speaking, the Arden Syntax is not a code system, but a language in which codes can be embedded. For example, Columbia Presbyterian Medical Center's MLM for epilepsy uses ICD-9 codes [ICD-9-CM97] to recognize the condition.

6.2.2. CPT4 (Current Procedural Terminology, 4th edition)

The Physicians' CPT [CPT498] is a listing of descriptive terms and identifying codes for reporting medical services and procedures performed by physicians. It is published annually by the American Medical Association. In the United States., CPT-4 is required by most payers for physician billing (e.g., health insurance companies), and is also required in addition to ICD-9 [ICD-9-CM97] for some technical billing.

The CPT coding scheme starts with six broad categories (Evaluation and Management, Anesthesiology, Surgery, Radiology, Pathology/Laboratory, and Medicine). Within these categories the codes are set out in an order that makes sense for that category. For example, anesthesiology codes are arranged by part of the body (head, neck, thorax, etc.), while medicine codes are arranged generally by specialty (ophthalmology, cardiovascular, pulmonary, etc.).

6.2.3. CYC

CYC [CYC98] is a very large, multi-contextual knowledge base and inference engine developed by Cycorp. The goal of the CYC project is to construct a foundation of basic "common sense" knowledge--a semantic substrate of terms, rules, and relations--that will enable a variety of knowledge-intensive products and services. CYC is intended to provide a "deep" layer of understanding that can be used by other programs to make them more flexible. A more detailed description of the CYC system can be found in Section 4.2.

Although CYC is capable of modeling just about any domain, a considerable amount of effort has gone into modeling concepts and assertions in the healthcare domain. The Cycorp staff includes experts in various healthcare fields to ensure that the healthcare-related concepts are accurately captured within the CYC system.

6.2.4. DRG (Diagnosis Related Group)

A DRG [DRG98] is a classification of a hospital stay in terms of what was wrong with, and what was done for, a patient. The DRG classification (one of about 500) is determined by a "grouper" program based on diagnoses and procedures coded in ICD-9 [ICD-9-CM97]), and on patient age, sex, length of stay, and other factors. The DRG frequently determines the amount of money that will be reimbursed, independently of the charges that the hospital may have incurred.

In the United States, the basic set of DRG codes are those defined for adult Medicare billing. For other patient types and payers - CHAMPUS (Civilian Health and Medical Services of the Uniformed Services), Medicaid, commercial payers for neonate claims, Workmans' Compensation - modified groupers and additional DRG codes are used.

6.2.5. ECRI's Medical Device Codes

ECRI (Emergency Care Research Institute) is an independent, non-profit institution that has provided the healthcare community, both nationally and internationally, with information about the safe and efficacious use of medical technology.

ECRI offers a set of codes for identifying medical equipment. The Health Devices Sourcebook, a directory file produced by ECRI, contains current address and marketing information on U.S. and Canadian manufacturers and distributors of more than 4,500 classes of medical devices. The database utilizes ECRI's internationally endorsed Universal Medical Device Nomenclature System (UMDNS) [UMDNS98]. An online thesaurus is available as an aid in locating broader, narrower, and related product names. The file contains three types of records: product (device) records, manufacturer records, and servicer records. International coverage of companies is provided.

6.2.6. ICD-9-CM (International Classification of Diseases, 9 th Revision, Clinical Modification)

The ICD-9-CM [ICD-9-CM97] is based on the official version of the World Health Organization's 9th Revision, International Classification of Diseases. ICD-9 is designed for the classification of morbidity and mortality information for statistical purposes and for the indexing of hospital records by disease and operations, for data storage and retrieval. Diagnoses and procedures coded in ICD-9-CM determine the DRG [DRG98] that controls reimbursement by U.S. Public Health Service and Health Care Financing Administration programs, and most other payers.

ICD-9 codes are sometimes thought to be better suited for billing than for categorizing clinical information. The fact remains however that every U.S. hospital's diagnoses and procedures code is in ICD-9-CM. As managed care erases the line between clinical and financial management, increasingly hospitals are mining ICD-coded records for the limited clinical information that may be found there.

6.2.7. ICD-10 (International Classification of Diseases, 10 th Revision)

ICD-10 [ICD-1098] is much larger than ICD-9. The number of categories available for the classification is significantly enlarged and further detail is available. The following is a quote from the Centers for Disease Control (CDC) Web site (http://www.cdc.gov) on how ICD-10-CM differs from ICD-9: " Notable improvements in the content and format include**:** the addition of information relevant to ambulatory and managed care encounters; expanded injury codes; the creation of combination diagnosis/symptoms codes to reduce the number of codes needed to fully describe a condition; the addition of a sixth character; incorporation of common 4th and 5th digit sub-classifications; laterality; and greater specificity in code assignment. The new structure will allow further expansion than was possible with ICD-9-CM."

At present ICD-10 is widely used in Europe. In the U.S., however, migration to ICD-10 is complicated by the fact that ICD-9-CM is embedded in hospital billing systems. The U.S. National Center for Health Statistics has developed a timeline recommended by the National Committee on Vital and Health Statistics (NCVHS) to have ICD-10-CM in use for morbidity diagnoses in the year 2001.

6.2.8. IUPAC (International Union of Pure and Applied Chemistry)

IUPAC [IUPAC98] is a voluntary non-governmental, non-profit organization that unites chemists from all over the world. The object of the Union is the advancement of both pure and applied chemistry.

IUPAC grew out of the international recognition of a need for standardization in chemistry, it being accepted that standardization of weights, measures, names and symbols is essential to the well-being and continued success of the scientific enterprise. Indeed, it is essential for the smooth development and growth of international trade and commerce.

It was this desire for international cooperation amongst chemists, and to facilitate the work of the international, but fragmented, chemistry community that was one of the earliest characteristics of the Union. Indeed, even before the creation of IUPAC (1919), the body out of which the Union developed, International Association of Chemical Societies (IACS), had met in Paris in 1911 and produced a set of proposals for the work that the new Association should address. These included:

- Nomenclature of inorganic and organic chemistry;
- Standardization of atomic weights;
- Standardization of physical constants;
- Editing tables of properties of matter;
- Establishing a commission for the review of work;
- Standardization of the formats of publications;
- Measures required to prevent repetition of the same papers.

6.2.9. LOINC (Logical Observation Identifiers, Names, and Codes)

The LOINC™ [LOINC98] database provides a set of universal names and ID codes for identifying laboratory and clinical observations. The purpose is to facilitate the exchange and pooling of clinical laboratory results, such as blood hemoglobin or serum potassium, for clinical care, outcome management, and research. Currently, many laboratories are using ASTM (American Society of Testing and Materials) 1238-94 or its sister standard, HL7, to send laboratory results electronically from producer laboratories to clinical care systems in hospitals. Most laboratories identify tests in these messages by means of their internal (and idiosyncratic) code values, so the receiving systems cannot fully "understand" the results they receive unless they either adopt the producer's laboratory codes (which is impossible if they receive results from multiple source laboratories, e.g.; the hospital lab, the local commercial lab, and a nursing home lab), or invest in work to map each laboratory's code system to the receiver's internal code system.

If laboratories all used the LOINC codes to identify their results in data transmissions, this problem would disappear. The receiving system with LOINC codes in its master vocabulary file would be able to understand and properly file HL7 (Health Level 7) results messages that also use the LOINC code. Similarly, government agencies would be able to pool results (within limits) for tests from many sites if they were reported electronically using the LOINC codes. The LOINC codes (and names) for test observations should be of interest to hospitals, clinical laboratories, doctors' offices, state health departments, governmental health care providers, third party payers, and organizations responsible for quality assurance and utilization review.

The LOINC codes are not intended to transmit all possible information about a test. They are only intended to identify the test result. Other fields in the message will transmit the identity of the source laboratory and the very detailed specimen information. (For instance, the code may identify a blood culture, but the message source code can be more specific and identify the specimen as “pumped blood”.) The level of detail in the LOINC definitions was intended to distinguish tests that are usually distinguished as separate test results within the master file of existing laboratory systems.

6.2.10. MeSH (Medical Subject Headings)

The Medical Subject Headings (MeSH) [MeSH98] comprise National Library of Medicine's controlled vocabulary used for indexing articles, for cataloging books and other holdings, and for searching MeSH-indexed databases.

MeSH terminology provides a consistent way to retrieve information that may use different terminology for the same concepts.

MeSH organizes its concepts in a hierarchical structure so searches for a broad concept may include articles indexed to narrower concepts. This structure also provides an effective way for searchers to browse MeSH in order to find appropriate concepts.

Subject specialists in various areas continually update the MeSH vocabulary. Each year hundreds of new concepts are added and thousands of modifications are made. The 1998 version of MeSH includes more than 18,000 main concepts and over 80,000 cross-references.

6.2.11. NANDA (North American Nursing Diagnosis Association)

NANDA [NANDA94] is a set of nursing diagnoses adopted by the North American Nursing Diagnosis Association. As compared to ICD-9-CM codes [ICD-9-CM97], which describe a disease or injury, NANDA nursing diagnoses describe a patient's reactions to the disease.

NANDA is a compact code set. The printed list takes up three to four pages. It is organized around nine "Human Response Patterns":

1. Exchanging
2. Communicating
3. Relating
4. Valuing
5. Choosing
6. Moving
7. Perceiving
8. Knowing
9. Feeling

Within each pattern, NANDA lists one to four subcategories. For example, under Exchanging, 1.3.2 is "altered urinary elimination," and 1.3.2.1.3 is "urge incontinence."

The American Nurses Association has adopted NANDA as the standard for nursing diagnoses in the U.S. NANDA is included in the National Library of Medicine UMLS Metathesaurus® [UMLS98].

6.2.12. NDC (National Drug Codes)

The National Drug Code (NDC) system [NDC98] identifies pharmaceuticals in great detail, even including the packaging. Its use is required by the U.S. Federal Drug Administration for reporting and it is used in many healthcare information systems. NCPDP (National Council for Prescription Drug Programs) drug transactions use NDC. At the end of 1995 there were over 170,000 NDC codes.

A fundamental weakness of the NDC system is that it has no reliable means to cross-link trade names with generics, or even different packages of the same medication. Multum (http://www.multum.com) offers a free database that maps NDC codes to clinically useful categories.

6.2.13. Read Codes

The Read Codes [READ98], developed in Great Britain, are a comprehensive list of terms intended for use by all healthcare professionals to describe the care and treatment of their patients. They enable the capture and retrieval of patient centered information in natural clinical language within computer systems.

The Read Codes cover such topics as occupations, signs and symptoms, investigations, diagnoses, treatments and therapies, drugs and appliances and much more. This enables the recording within the computer system of anything from a summary of the episode of care to potentially a full electronic patient record if desired.

Each term is attached to a unique Read Code that is stored within the computer. This allows storage, retrieval and analysis of data. When the information is returned to the screen, the clinician is presented not with the code, but with the familiar clinical term, which therefore retains its usefulness.

The Read Codes can make patient records easily retrievable. The computerized record structure can be accessed both for individual patient care and for purposes such as running practice functions, and in secondary care crossmapping to ICD-10 [ICD1098].

6.2.14. SNOMED (Systemized Nomenclature of Human and Veterinary Medicine)

SNOMED [SNOMED86] was conceived from the start as a system for representing clinical information. Unlike ICD-9 and CPT4, it is not used for billing in the U.S.

Dr. Roger A. Cote, principal author of SNOMED, contrasts it to ICD [ICD-9-CM97]: while the latter is a classification, SNOMED is a "multi-axial coded medical nomenclature." A number of separate codes, one per axis, might constitute a diagnosis. For example, lung (topographical axis) plus granuloma (morphological axis) plus fever (functional axis) plus myobaterium tuberculosis (etiology axis) might add up to a diagnosis of tuberculosis (disease axis).

SNOMED was introduced in September 1993 and is traceable to its roots in the early 1960s as SNOP – the Systematized Nomenclature for Pathology. The new (current version 3.2) of SNOMED International is a comprehensive, multi-axial nomenclature classification work created for the indexing of the entire medical record, including signs and symptoms, diagnoses, and procedures. Its unique design will allow full integration of all medical information in the electronic medical record into a single data structure.

The most current version (June 1996) contains more than 138,000 terms and term codes in 11 separate Modules. The College of American Pathologists and its SNOMED Editorial Board are committed to providing updates at least twice each year.

SNOMED International is rapidly being accepted worldwide (it is being translated into 13 languages other than English) as the standard for indexing medical record information. The American Veterinary Medical Association and the American Dental Association have recognized SNOMED's virtues and have adopted/endorsed SNOMED for their use. In addition, the American College of Radiology/National Equipment Manufacturers Association will be using a subset of SNOMED in their Digital Imaging Communications Standard (DICOM).

6.3. UMLS

6.3.1. Overview

UMLS [UMLS98] helps health professionals and researchers retrieve and integrate electronic biomedical information from a variety of sources. It can be used to overcome variations in the way similar concepts are expressed in different sources. This makes it easier for users to link information from patient record systems, bibliographic databases, factual databases, expert systems, etc. The UMLS Knowledge Services can also assist in data creation and indexing applications.

The UMLS includes machine-readable "Knowledge Sources" that can be used by a wide variety of applications programs to compensate for differences in the way concepts are expressed in different machine-readable sources and by different users, to identify the information sources most relevant to a user inquiry. The Metathesaurus contains mappings to MeSH (Medical Subject Headings at the National Library of Medicine) [MeSH98], ICD-9-CM [ICD-9-CM97], SNOMED [SNOMED86], CPT [CPT498], and a number of other coding systems.

6.3.2. Parts of UMLS

There are four UMLS knowledge sources, the Metathesaurus, Semantic Network, SPECIALIST Lexicon, and Information Sources Map. Because the first two knowledge sources are of most interest to this paper, they are described in greater detail below.

6.3.2.1. Metathesaurus

The Metathesaurus contains information about biomedical concepts and terms from many controlled vocabularies and classifications used in patient records, administrative health data, bibliographic and full-text databases and expert systems. It preserves the names, meanings, hierarchical contexts, attributes, and inter-term relationships present in its source vocabularies; adds certain basic information to each concept; and establishes new relationships between terms from different source vocabularies.

The Metathesaurus supplies information that computer programs can use to interpret user inquiries, interact with users to refine their questions, identify which databases contain information relevant to particular inquiries, and convert the users' terms into the vocabulary used by relevant information sources. The scope of the Metathesaurus is determined by the combined scope of its source vocabularies. The Metathesaurus is produced by automated processing of machine-readable versions of its source vocabularies, followed by human review and editing by subject experts. The Metathesaurus is intended primarily for use by system developers, but can also be a useful reference tool for database builders, librarians, and other information professionals.

The Metathesaurus is organized by concept or meaning. Alternate names for the same concept (synonyms, lexical variants, and translations) are linked together. Each Metathesaurus concept has attributes that help to define its meaning, e.g., the semantic type(s) or categories to which it belongs, its position in the hierarchical contexts from various source vocabularies, and, for many concepts, a definition. A number of relationships between different concepts are represented. Some of these relationships are derived from the source vocabularies; others are created during the construction of the Metathesaurus. Most inter-concept relationships in the Metathesaurus link concepts that are similar along some dimension.

6.3.2.2. Semantics Network

The Semantic Network, through its 132 semantic types, provides a consistent categorization of all concepts represented in the UMLS Metathesaurus. The 53 links among the semantic types provide the structure for the Network and represent important relationships in the biomedical domain. All information about specific concepts is found in the Metathesaurus; the Network provides information about the basic semantic types that are assigned to these concepts, and it defines the relationships that may hold between the semantic types.

The Semantic Network serves as an authority for the semantic types that are assigned to concepts in the Metathesaurus and that are assigned to databases in the Information Sources Map (ISM). The Network defines these types, both with textual descriptions and by means of the information inherent in its hierarchies.

The semantic types are the nodes in the Network, and the relationships among them are the links. There are major groupings of semantic types for organisms, anatomical structures, biologic function, chemicals, events, physical objects, and concepts or ideas. The current scope of the UMLS semantic types is broad, allowing for the semantic categorization of a wide range of terminology in multiple domains.

The primary link is the `isa' link. This establishes the hierarchy of types within the Network and is used for deciding on the most specific semantic type available for assignment to a Metathesaurus concept. In addition, a set of non-hierarchical relations between the types has been identified. These are grouped into five major categories, which are themselves relations: `physically related to,' `spatially related to,' `temporally related to,' `functionally related to,' and `conceptually related to.'

6.3.2.3. Specialist Lexicon

The SPECIALIST lexicon is an English language lexicon with many biomedical terms. It has been developed in the context of the SPECIALIST natural language processing project at National Library of Medicine. The current version includes some 108,000 lexical records, with over 186,000 strings.

The lexicon entry for each word or term records syntactic, morphological, and orthographic information. Lexical entries may be single or multi-word terms. Entries that share their base form and spelling variants, if any, are collected into a single lexical record. The base form is the uninflected form of the lexical item; the singular form in the case of a noun, the infinitive form in the case of a verb, and the positive form in the case of an adjective or adverb.

6.3.2.4. Information Sources Map (ISM)

The problem of identifying appropriate information resources to answer a specific question is as old as librarianship itself. It has gained in complexity with the proliferation of printed and electronic information. The latter, when remotely located, has in the past been accessed mainly via modem-based telephone connections. In recent years, electronic information resources have been increasingly accessed via the rapidly growing global computer communications network known as the Internet, and in particular via the Internet's most popular information retrieval system, World Wide Web (WWW).

The ISM is a database that describes information sources. Software tools being developed in conjunction with the ISM aim to support the following capabilities:

- Determine which information sources are most likely to be relevant to a particular query.
- Supply human-readable information about the scope, probable utility, and access conditions of identified sources.
- Automatically connect to the identified information sources, retrieving and organizing appropriate information.

6.4. Observations

The challenge that the healthcare community is trying to tackle through the use of the ontologies in the previous section is not much different than the problems that the manufacturing community is facing; namely, the integration of previously isolated knowledge sources. Another similarity is the way they are trying to use ontologies to solve their problem. The third approach listed in Section 5.1.3, using a single ontological-based architecture to bring together various knowledge sources in an single “enterprise model” (to use manufacturing terminology) is the approach that UMLS is using and which seems to be the most appropriate mechanism for the healthcare community. UMLS can provide the many pieces of healthcare-related information necessary to aid in making appropriate decisions and facilitate this decision-making process.

One difference between the use of ontologies in the healthcare and manufacturing domains is the need for inferencing. It seems that in manufacturing there is a much greater need to be able to deduce new information from existing information as oppose to healthcare in which the clear representation of the information is important. This, of course, could change over time.

7.0 Conclusion/Future Work

Not surprisingly, the analysis of the various ontological systems has shown that there are numerous approaches and methodologies to formally modeling information. Many of these modeling decisions seem to be focused around the intended use of the ontology once created, with very few of these ontologies being created for general use (with the possible exception of the future Ad Hoc Ontology standard [ANSI98]). For this reason, the analysis of these systems with respect to modeling manufacturing information showed that each of these systems offered its own, unique advantages. Because the analysis focused on three main criteria – the ability to represent manufacturing information, the amount of manufacturing information that was already represented, and the ability to inference over the information represented – the project participants decided to proceed with the CYC system.

With the identification of an appropriate ontology complete, the project will move on to develop a manufacturing taxonomy and ontology using the chosen system. This will involve at least the following three steps:

1. **Identification and definition of domain-specific concepts**
   There have been many efforts to identify semantic concepts in specific domains within manufacturing (e.g., process planning, design, product). The precise definitions of these concepts will be determined and documented to serve as a basis for the ontology described in the next phase.
2. **Ontology population**
   With the knowledge gained from training sessions, work will be performed to model the concepts in the previous phase in the chosen ontological system. At first one domain will be modeled and issues will be tracked and resolved. These lessons learned will then help to guide the modeling of other domains. The output of this work will be the beginning of a domain-specific taxonomy.
3. **Domain ontology merging and determination and resolution of semantic mismatches**
   As work on the ontology creation continues, it is fully expected that there will be conflicting definitions of the same term in different domains. For example, a resource in one domain may be completely different than a resource in another domain. During this phase, these differences will be studied and resolved, possibly through the generalization or specialization of necessary concepts. These various domain ontologies will be merged which will result in a common taxonomy.

8.0 References

[deK86] deKleer, Johan, "An Assumption-Based Truth Maintenance System" Artificial Intelligence Journal 28 (163-196), Deerborn, MI, 1986.

[ANSI98] ANSI Ad Hoc Group on Ontology Standards, http://www.ksl.stanford.edu/, August 20, 1998.

[ARDEN92] American Society for Testing and Materials. E 1460 Standard Specification for Defining And Sharing Modular Health Knowledge Bases (Arden Syntax for Medical Logic Modules). ASTM Standards, v 14.01. Philadelphia: ASTM, 1992; 539-87.

[BLO33] Bloomfield, Leonard, "Language" University of Chicago Press, 1933.

[BUI98] "Building bug-free O-O software: An introduction to Design by Contract(TM)," http://www.eiffel.com/doc/manuals/technology/contract/index.html, August 27, 1998.

[CAMI91] McKay, Ken, et. al., "Report: Intelligent Manufacturing Management Program, State of the Art Scheduling Survey," CAM-I Report R-91-IMM-01, Waterloo, Ontario, Canada, 1991.

[CHA97] Chaudhri, V., A. Farquhar, R. Fikes, P. Karp and J. Rice, *The Generic Frame Protocol 2.0*, KSL-97-05 (technical report), Stanford Knowledge Systems Laboratory, Stanford University, Stanford, CA, 1997.

[CPT498] American Medical Association, Physicians' Current Procedural Terminology: CPT 4th Edition, Chicago, IL, 1998.

[CYC98] Cycorp Inc, The CYC Technology, http://www.cyc.com/tech.html, July 31, 1998.

[DAN97] Danner, William F., "Developing Application Protocols (APs) Using the architecture and methods of STEP (STandard for the Exchange of Product data) Fundamentals of the STEP Methodology. NIST Internal Report (NISTIR) 5972, January 1997.

[DRG98] Duke University Medical Center, Diagnosis Related Group (DRG), http://www.mcis.duke.edu/standards/termcode/drg.htm, July 31, 1998.

[ENT98] Mike Uschold, Martin King, Stuart Moralee and Yannis Zorgios, "The Enterprise Ontology," The Knowledge Engineering Review, Vol. 13, Special Issue on Putting Ontologies to Use (eds. Mike Uschold and Austin Tate), 1998.

[FAR96] Farquhar, A., R. Fikes and J. Rice, "The Ontolingua Server: A Tool for Collaborative Ontology Construction," *Proceedings of the Tenth Knowledge Acquisition for Knowledge-Based Systems Workshop*, Banff, Canada, November 1996.

[GEN90] Genesereth, M. R., *The Epikit Manual*, Epistemics, Inc., Palo Alto, CA, 1990.

[GEN92] Genesereth, M. R. and R. E. Fikes, *Knowledge Interchange Format, Version 3.0 Reference Manual*, KSL-92-86 (technical report), Stanford Knowledge Systems Laboratory, Stanford University, Stanford, CA, 1992.

[GRU92] Gruber, T. R., *Ontolingua: A Mechanism to Support Portable Ontologies*, KSL-91-66 (technical report), Stanford Knowledge Systems Laboratory, Stanford University, Stanford, CA, 1992.

[ICD-9-CM97] Healthcare Financing Administration, The International Classification of Diseases: 9th Revision, Clinical Modification: ICD-9-CM, 1997.

[ICD-1098] International Classification of Diseases and Related Health Problems, Tenth Revision, Health Care Financing Administration, Washington DC, 1998.

[ISO94] ISO, 10303-1:1994, Product data representation and exchange: Part 1: overview and fundamental principles.

[ISO97] ISO/IEC 61360-4:1997 Standard data element types with associated classification scheme for electric components. http://www.codus.co.uk/intstds.htm

[IUPAC98] Duke University Medical Center, International Union of Pure and Applied Chemistry, http://www.mcis.duke.edu/standards/termcode/iupac.htm, July 31, 1998.

[JEZ98] Jezequel, Jean-Marc and Meyer, Bertrand, "Put it in the contract: The lessons of Ariane" http://www.eiffel.de/pages/misc/ariane.htm, August 27, 1998.

[LOINC98] Duke University Medical Center, LOINC™ File Access, http://www.mcis.duke.edu/standards/termcode/loinclab/loinc.html, July 31, 1998.

[LOOM98] "Loom Homepage" http://www.isi.edu/isd/LOOM/LOOM-HOME.html, August 20, 1998.

[MAC91] MacGregor, R., "The Evolving Technology of Classification-based Knowledge Representation Systems," *Principles of Semantic Networks: Explorations in the Representation of Knowledge*, J. Sowa (ed.), Morgan Kaufmann, San Mateo, CA, 1991.

[MESH98] National Library of Medicine, Medical Subject Heading (MeSH), Bethesda, MD, 1998.

[MIKRO95] Mahesh, K., and Nirenburg, S., A Situated Ontology for Practical NLP. In Proc. Workshop on Basic Ontological Issues in Knowledge Sharing, International Joint Conference on Artificial Intelligence (IJCAI-95), Montreal, Canada, Aug. 19-20, 1995.

[NANDA94] Carroll-Johnson, Rose Mary, editor, Classification of Nursing Diagnosis: Proceedings of the 10th Conference of Nursing Diagnosis, 1994.

[NDC98] Duke University Medical Center, National Drug Codes (NDC), http://www.mcis.duke.edu/standards/termcode/ndc.htm, July 31, 1998.

[OMG98] OMG (Object Management Group) http://www.omg.org/, August 27, 1998.

[ONTO96] A. Farquhar, R. Fikes, & J. Rice. The Ontolingua Server: A Tool for Collaborative Ontology Construction. Knowledge Systems Laboratory, KSL-96-26, Stanford, CA, September 1996.

[QUI87] Quine, W.V., "Quiddities: An Intermittently Philosophical Dictionary", The Belknap Press of Harvard University Press, Cambridge, Massachusetts, 1987.

[READ98] Computer Aided Medical Systems (CAMS) Limited, CAMS Online – The Homepage of Read Codes, http://www.cams.co.uk/index.htm, July 31, 1998.

[SCH81] Schank, Roger C., and Riesbeck, Christopher K., "Inside Computer Understanding" Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1981.

[SENS98] "Ontology Creation and Use – SENSUS," Natural Language Group at USC/ISI, http://www.isi.edu/natural-langauge/resources/sensus.html, November 30, 1998.

[SNOMED86] Cote, R. A., Architecture of SNOMED. Its Contribution to Medical Language Processing. Proceedings of the Tenth Annual Symposium on Computer Applications in Medical Care. Edited by Helmuth D. Orthner, Computer Society of the IEEE, Computer Society Press, Northfield, IL, 1986.

[SPAR98] Tate, Austin, "Planning Initiative: Shared Planning and Activity Representation: SPAR Homepage", http://www.aiai.ed.ac.uk/~arpi/spar/, November 30, 1998.

[TOVE98] Gruninger, Michael, "Enterprise Modeling," http://www.ie.utoronto.ca/EIL/comsen.html, August 19, 1998.

[UMLS98] National Library if Medicine, UMLS Knowledge Sources User's Guide, 9th Edition, Bethesda, MD, January 1998.

[UMDNS98] ECRI, Universal Medical Devices Nomenclature System; Product Category Thesaurus, Plymouth Meeting, PA, 1998.

[USH98] Uschold, M., M. King, S. Moralee and Y. Zorgios, "The Enterprise Ontology," *The Knowledge Engineering Review* , **13,** Edinburgh, Scotland, 1998.

[VAR91] Varela, Francisco J., Thompson, Evan and Rosch, Elanor, "The Embodied Mind: Cognitive Science and Human Experience" The MIT Press, Cambridge, Massachusetts, 1991.

[WORD98] "Wordnet **--** A Lexical Database for English", Cognitive Science Laboratory, Princeton University, http://www.cogsci.princeton.edu/~wn/, November 30, 1998.

[XML98] Extensible Markup Language (XML) 1.0: W3C Recommendation from 10-February-1998, http://www.w3.org/TR/REC-xml, August 28, 1998.