A Measurement Ontology Generalizable for Emerging Domain Applications on the Semantic Web

Henry M. Kim, York University, Canada

Arijit Sengupta, Wright State University, USA

Mark S. Fox, University of Toronto, Canada

Mehmet Dalkilic, Indiana University, USA

ABSTRACT

This article introduces a measurement ontology for applications to Semantic Web applications, specifically for emerging domains such as microarray analysis. The Semantic Web is the next-generation Web of structured data that are automatically shared by software agents, which apply definitions and constraints organized in ontologies to correctly process data from disparate sources. One facet needed to develop Semantic Web ontologies of emerging domains is creating ontologies of concepts that are common to these domains. These general "common-sense" ontologies can be used as building blocks to develop more domain-specific ontologies. However most measurement ontologies concentrate on representing units of measurement and quantities, and not on other measurement concepts such as sampling, mean values, and evaluations of quality based on measurements. In this article, we elaborate on a measurement ontology that represents all these concepts. We present the generality of the ontology, and describe how it is developed, used for analysis and validated.

Keywords: measurement; microarray analysis; ontological engineering; ontology; quality control; quality of service (QoS); Semantic Web

INTRODUCTION

According to Tim Berners-Lee, whom many attribute as the inventor of the World Wide Web, the Web will evolve into the *Semantic Web*, which relies upon using machine processable domain knowledge represented in *ontologies* to execute and compose automated *Web services* (Berners-Lee, Hendler, & Lassila, 2001; Chen, Zhou, & Zhang, 2006). An ontology is a data model that "consists of a representational vocabulary with precise definitions of the meanings of the terms of this vocabulary plus a set of formal axioms that constrain

interpretation and well-formed use of these terms" (Campbell & Shapiro, 1995). Ontology use ensures that data instances are so precisely defined and constrained that the instances can be processed automatically and accurately by Web-based computer programs, or *software agents*. Berners-Lee's et. al (2001) vision of the Semantic Web is that "many software agents, accessing data instances and applying ontologies to the instances, execute Web services in concert, where agents, data instances, and ontologies are distributed all over the Web."

Ontologies for the Semantic Web represent an emerging method for modeling the semantics required to interpret data. In a similar vein, applications such as genomics and GIS's represent emerging domains represented for semantic modeling. There exist Semantic Web ontologies for traditional applications in computer science, and business (Davies, Duke, & Stonkus, 2002; Gandon & Sadeh, 2004; Klischewski & Jeenicke, 2004). There are also emerging domains modeled using traditional semantics modeling techniques (Khatri, Ram, & Snodgrass, 2004; Ram & Wei, 2004). There are even some ontologies of emerging domains such as representations of the Gene Ontology (Ashburner et al., 2000; Wroe, Stevens, Goble, & Ashburner, 2003) and a bioinformatics ontology (Stevens, Goble, Horrocks, & Bechhofer, 2002), which are represented in the de facto Semantic Web Ontology Language, OWL (McGuinness & van Harmelen, 2003), or its predecessor, DAML+OIL (Bechhofer, Goble, & Horrocks, 2001). Although one of the goals of ontology development is a generalization of terms in an application, it is possible to make intelligent choices when several ontologies are available for the same domain (Lozano-Tello & Gomez-Perez, 2004).

When contemplating the development of ontologies of any domain, it is instructive to state the following informal definition: "an ontology is an explicit representation of shared understanding" (Gruber, 1993). Gruber also outlines the conundrum of ontological commitment: the more one commits to represent a given domain in an ontology to make data more sharable for software applications closely associated with that domain's needs, the less sharable the data becomes for other applications. For example, commitments made to develop a gene ontology useful for genomics applications render the ontology to be less likely to be used to share medical records data. Yet data sharing between genomics and patient care applications is critical in many situations. The remedy to this seeming conundrum is to identify general concepts that cut across many domains—the domains' "common-sense"—and collect them in a common-sense ontology (Lenat, 1995; Milton & Kazmierczak, 2004). This ontology is separated from more domain-specific ones. In fact, terms in several domain-specific ontologies can be defined using terms from a common general ontology. For example, a molecular biological ontology may provide building block representations for a biomedical ontology, which in turn underpins both gene and health care ontologies. The general ontologies also underlie an ontology of a different perspective—that of costing (Fox & Gruninger, 1998).

Arguably the ontologies of *emerging domains*, to be discussed below, make ontological commitments to their respective domains. It is prudent to ask, however, what are the common-sense ontologies that underlie these emerging domains? Can their representations serve as building blocks to define Semantic Web ontologies of different

emerging domains? The attributes shared among common emerging domains, which sets them apart from traditional business and computer science domains, is their grounding in the physical sciences. The common-sense of emerging domains is the common-sense of our physical world. There are common-sense models of time (Allen, 1983), space (Retz-Schmidt, 1988), causality (Reiger & Grinberg, 1977), and physics (Hayes, 1985; Kuipers, 1986). So how do traditional and emerging domains differ? We set our sights on the life sciences, since it is here that the most significant number of emerging technologies have taken place.

The life sciences have relied on the so-called scientific method—discovery through a process of observation, hypothesis formulation and data generation. In these traditional domains, data play an ancillary role to the hypotheses (these domains are often referred to as "hypothesis driven"). For example, a scientist believes that two particular genes might have some kind of relationship in some stage of larval development in a *D. melanogaster*. Experiments are then conducted to validate this conjecture. The data are generally interesting only in the context of this experiment. There are long standing traditional as well as practical reasons (Attie, 2003) why data is put into this ancillary role, and we only present a few here: they need not be present during the observation and hypothesis formulation phases of discovery; furthermore, a surplus of data does usually little to enhance this process of discovery; data are often prohibitively expensive to produce or gather once, let alone many times. Emerging domains (Ronco, Grossel, Zimmer, & Socash, 2003), like bioinformatics (and particular areas of focus, for example, genomics, proteomics, metabolomics) through a direct conse-

quence of recent and rapid advancements in technology, have taken the scientific method and made it, in a sense, stand on its head. There are also domains where data are generated and discoveries made without any preceding hypothesis (Attie, 2003) and are often referred to as "technologically driven." Technology in this case typically refers to high-throughput gene product data where tens of thousands of interactions are measured in unison. To motivate our article, we will focus on microarrays (Duggan, Chen, Meltzer, & Trent, 1999; Mount, 2004), the most well-known and ubiquitous of the technologically driven data tools. Numerous challenges are faced by these emerging disciplines. One critical problem is data quality—many of the high throughput techniques, microarrays included, suffer from a great deal of noise (Lesk, 2005). Furthermore, no formal ANSI standards exist for managing these data systematically though some informal standards have emerged. One such standard is the MIAME format for microarrays (Duggan, Chen, Meltzer, & Trent, 1999; Mount, 2004), though this format does not include quality control information. So, handling data quality is made even more difficult and can be cast into a more general framework of measurement.

There is the abstract and mathematical field of measurement theory (Roberts, 1979), which serves as a valuable reference for those works that represent measurement concepts in an ontology. Whether stated in formal data models (Goh, Bressan, Madnick, & Siegel, 1999; Novak, 1995), or as computational ontologies (Gruber & Olsen, 1994), works that can be considered ontologies of measurement are predominantly concerned with units of measurement, specifically using the ontology to accurately convert from one unit of measure to another.

This holds for Semantic Web measurement ontologies for traditional domains; some examples are the measurement ontologies for KSL (Fikes & Farquhar, 1999), SHOE (Heflin, 2000), Cyc® (Lenat, 1995), SUMO (Pease & Niles, 2002) and GNU (Nozick, Turnquist, Jones, Davis, & Lawton, 2004) projects. The same can be said for emerging domains, and examples include the measurement units represented in SEEK (ecology) (Bowers & Ludäscher, 2004), MGED (genomics) (Stoeckert Jr. & Parkinson, 2004), and PharmGKb (pharmacology) (Hewett et al., 2002) ontologies.

Measurement theory is certainly more than just measurement unit conversions and quantities. Therefore, regardless of whether it is for traditional or emerging domains, or whether it is for the Semantic Web or closed networks, there is a definite research opportunity in developing an ontology that represents additional measurement common-sense. With such an ontology in tow, developers of Semantic Web ontologies of emerging domains as diverse as biology and geography have a reference for ensuring that important and subtle measurement concepts are diligently represented in their ontologies and not ignored. *Ecolingua* is an ontology for modeling ecological metadata (Brilhante & Robertson, 2001) and represents the important measurement notion of sampling. What is needed is a measurement ontology that, like *Ecolingua*, can be used for developing Semantic Web ontologies of emerging domains, but represents even more.

We believe that the TOVE measurement ontology (Fox, 1992) is one of the most extensive ontologies for enterprise modeling (Noy & Hafner, 1997). This article elaborates on what had previously been written about this measurement ontology.

We first provide two scenarios for ontology use: one for the emerging domain of genomics using microarrays and the other for the traditional domain of enterprise applications. We then present a demonstration of how the ontology can be used for this scenario. The demonstration has two parallel audiences. We show how the ontology can be used by a scientist who is working in one of these emerging domains, e.g., a bioinformatician who seeks to discover significant gene-gene interactions in the presence of noise, missing data, and conflicting data. We further demonstrate that this can transparently be cast in terms of an enterprise domain. We present arguments on generalizablity of the ontology. Finally, we make concluding remarks and state our intentions for future work.

MEASUREMENT ONTOLOGY

Finkelstein (1984) defines measurement as the process by which numbers or symbols are assigned to attributes of entities in the real world in such a way as to describe them according to clearly defined rules. In a philosophical discussion of an ontology of measurement units, Mars (1993) summarizes the modern axiomatic treatment of the theory of mathematics (Massey, 1971; Narens, 1985) as the problem of finding an axiomatic introduction of a function φ that assigns a numerical value to an object. Axioms of (1) ordering ($a \gtrsim b \leftrightarrow \varphi(a) \geq \varphi(b)$, where $\gtrsim$ denotes an empirical ordering of objects), and (2) extension ($\varphi(a) + \varphi(b) = \varphi(a \cdot b)$, where • denotes an empirical combination of two objects) are central. These axioms are formal and sharable, since they are expressed in mathematics, and represent fundamental measurement concepts. Fenton (1994) relates measurement as fundamental for ensuring software quality,

and gives practical guidance in delineating the fundamentals of measurement into "what is measurement" (e.g., types: direct vs. indirect) and "issues for measurement" (e.g., which type of attributes can and cannot be measured, and on what kinds of scales). We synthesize these concepts using the motivation of a real-life enterprise application to transform abstract concepts to explicit ontology terms, definitions, and constraints.

A graphical overview of the ontology development steps in the methodology that Gruninger and Fox (1995) used to engineer the TOVE Measurement Ontology representations is shown in Figure 1. These steps involve (1) developing a *motivating scenario*, (2) designing *informal competency questions* to capture the scenario, (3) formalizing them to more *formal competency questions*, and (4) logically deducing the competency questions as a *demonstration of competency*.

A *motivating scenario* is a detailed narrative about a specific enterprise, where emphasis is placed on problems or tasks

it faces. When the scenario is analyzed, enterprise-independent, generic concepts are abstracted to state *informal competency questions* in natural language *that bind the query capability or competency of the ontology*. Terms with which such queries can be composed comprise the *terminology* or data model of the ontology. Queries restated using the terminology and restrictive syntax of the ontology are called *formal competency questions*. Answers to these questions can be logically deduced if *axioms* that define and constrain the terminology are developed as restrictive semantics. These deductions constitute a *demonstration of competency*.

This methodology, which has posing and answering of informal and formal competency questions as its basis, is arguably the most popular ontology development methodology. According to a search via Google Scholar™ (www.scholar.google. com) in July 2005, Uschold and Gruninger's (1996) paper on ontological engineering is the fifth most cited paper on computational ontologies, and by far the most popular

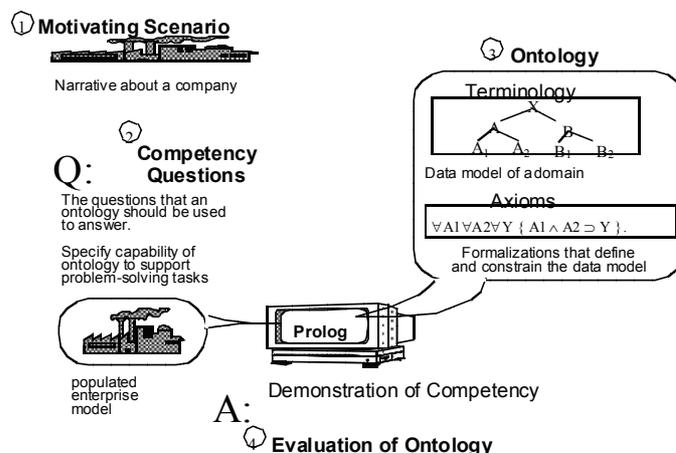*Figure 1. TOVE ontological engineering methodology*

Figure 2. This graph depicts gene/gene relationships that have been experimentally determined (lines—thickness is strength). The two circles are uncharacterized genes. The small shapes are further annotations of information to help the emerging domain scientist explore possible characterizations of the two uncharacterized ones.
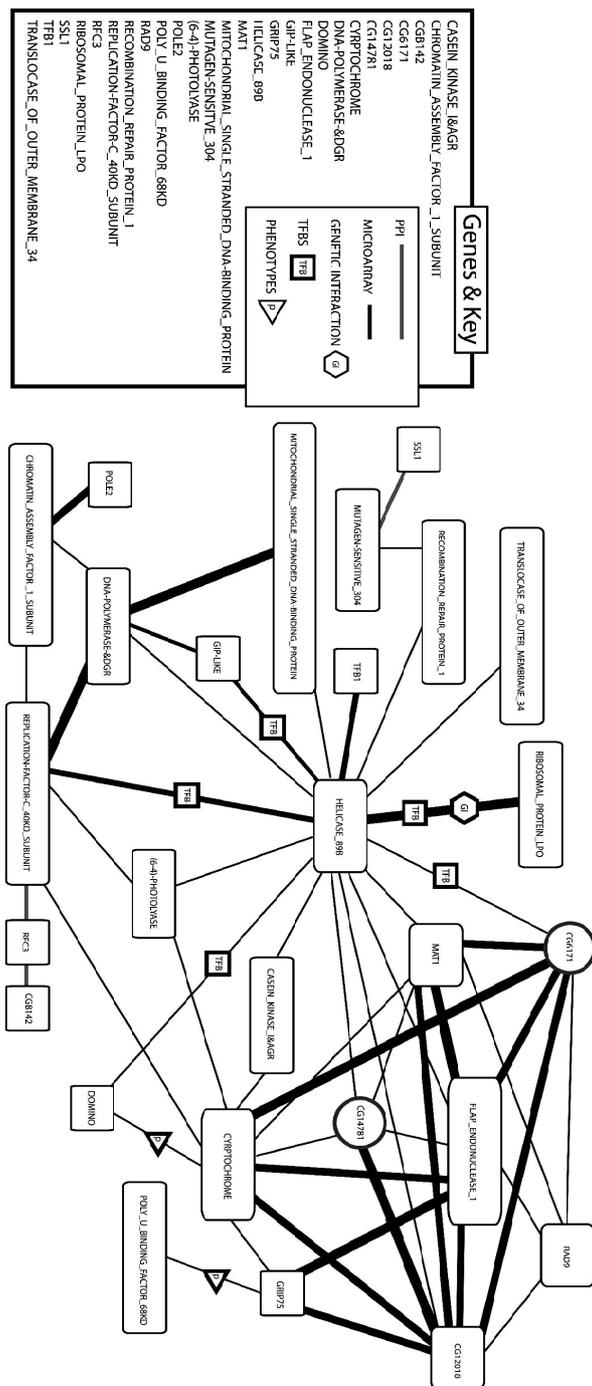
paper on ontology development methodology. This methodology has been applied to develop ontologies for fields as diverse as knowledge management (Gandon, 2001; Kim, 2002), bioinformatics (Stevens, Goble, & Bechhofer, 2000) and transportation engineering (Schlenoff, Balakirsky, Uschold, Provine, & Smith, 2003). These examples provide ample evidence that the methodology used to develop the measurement ontology can be used to develop ontologies of a wide variety of domains beyond enterprise modeling, the domain from which the ontology originates.

Motivating Scenario 1:
Gene Expression Microarrays

Functional genomics is the study of gene function and their respective relationships among each other. A multitude of perspectives exist as to the kinds of relationships—phenotypic expression, protein-protein interaction, but microarrays measuring gene expression levels are most commonly used. A number of promising and potential discoveries in functional genomics can be made that include drug discovery, biomarker determination, and biodiversity. We will present a high level description of microarrays, though a detailed description can be found in Mount (2004).

To identify the primary concepts for the measurement aspects of microarrays, we consider some of the details of microarray analysis:

Microarrays measure levels of associated substrate: one chemical species is fixed to a medium—a slide—and collections of other chemical species, that are fluorescently labeled, are brought into contact with the first. Those most strongly attracted adhere—called hybridization—and are then examined to provide some kind of degree of affinity by measuring the light intensities. Controls are added to give baseline indications of affinity and trials are repeated to help establish the quality of results. Recent technological advances have made it possible to measure thousands of these relationships on a single slide.

The key concept (C-1) abstracted from this excerpt is the following:

C-1: *There must be a systematic way of describing how a particular physical characteristic is to be measured and this description must be used to meet the quality expectations of the scientists.*

To obtain the next concept, we now consider some of the activities that are involved in microarray analysis.

There are a number of standard activities that take place when conducting microarray experiments. The substrates must be decided upon and procured, then fixed to the slides. The possible other substrates must be prepared by fluorescent labeling before hybridization. After exposure the slides must be washed of unbound substrates, scanned, and intensities displayed numerically.

The concept that can be derived from the above excerpt is the following:

C-2: *Quality assessment is made through a system of activities, which perform measurement; this is a view of measurement as an activity.*

Finally, we consider the decision process using the measurements:

Used cleverly, many kinds of relationships can be discovered. For example, given a set of genes, we may want to decide if significant interactions exist among them. In the case of functional genomics, gene-gene interactions can be indirectly observed by measuring the amounts of RNA that bind to DNA fragments. Gene expression can be identified with one of four states: up-regulated (enhanced production), equally expressed, no observable change, and down-regulated (inhibited production). Historically, green, black, yellow, and red have been used, respectively, with these four states where various intermediary colors represent a degree of mixing of these states. The final product then is a slide of a rectangular collection of tiny dots of color. These dots of color are actually of ratio of probes labeled with two fluorophores, commonly cyanine 3 (Cy3) peak absorption at 550 nm and emission at 570 nm and cyanine 5 (Cy5) with peak absorption at 649 and emission at 670 nm. The ratio is Cy5:Cy3, where equal intensities of both shows yellow, Cy5 > Cy3 shows red, Cy5 < Cy3 shows green, and black means neither is expressed. Microarrays are subject to numerous statistical variations brought about not only by the complex nature of the technology itself, but also by the sheer number of things being measured. To compare multiple run values must be further processed by, for example, normalization and include dimensions of time, light intensity, treatments, etc. To be at all useful, the quality of output must be well-known. Several methods are applied for improving quality, including sampling and repeated measurement of the same cell (repeat runs).

The concept embedded in the decision making process can be simply stated as:

C-3 Every quality assessment is a decision that begins with a value of measurement at a given point in time.

Motivating Scenario 2: BHP Steel

To show the applicability of the concepts, we take an alternative scenario from a more traditional standpoint of enterprise measurement.

BHP Steel is an industrial collaborator for TOVE Measurement Ontology development. The following excerpt describes its losses with respect to cost, time, and revenue when products of unacceptable quality (called non-prime products) are produced.

As raw materials are transformed by the different production units of BHP Steel's supply chain, non-prime products may be produced. These are the products whose physical properties do not satisfy necessary tolerance specifications. Non-prime products lead to lost revenue due to re-grading and scrapping, increased costs due to additional rework, carrying of excess inventory to meet delivery promises, and increased variability of leadtime performance.

The same key concept (C-1) can be abstracted from this excerpt:

C-4 There must be a systematic way of describing how a particular physical characteristic is to be measured and this description must be used to meet the customer expectations of quality.

The next excerpt describes BHP Steel's need to understand and improve its inspection processes, the collection of activities that assesses whether a product is non-prime.

If the products are consistently found to be non-prime, this is an indication that there is something faulty in the production unit. A cause for this occurrence is suspected to be an inadequate inspection processes.

C-5: *Quality assessment is made through a system of activities, which perform measurement.*

The following excerpt specifies what is entailed in determining a product as non-prime.

Especially when the product is shipped to the customer, it is essential that the product satisfies the tolerance specifications of the customer. Therefore, the product's physical characteristics are measured, compared against tolerance specifications and a decision about whether the product is non-prime is made.

C-6: *Every quality assessment uses a decision process that begins with a value of measurement at a given point in time.*

Informal Competency Questions

Not only are the informal competency questions developed to elaborate concepts, they are also designed from the findings of the measurement models' review.

Measurement Description System: To elaborate C-1, the transformation of the relationship between an entity and its attributes into the more tractable domain of term, numbers, and operators must be modeled (Grady, 1993). The following then are informal competency questions (e.g., ICQ-1) about requirements:

- **ICQ-1:** Is this a quality requirement?
- **ICQ-2:** What are the physical characteristics that are measured?

In measuring physical characteristics, one important aspect is sampling, which occurs when a subset of a population of an evaluated entity is measured, rather than the whole (Scheaffer & McClave, 1982). The following are some questions for representing sampling.

- **ICQ-3:** Is every entity that is produced measured?
- **ICQ-4:** If the product is a batch, is a sample taken from that batch and measured?
- **ICQ-5:** If a sample is taken and measured, is the value for the measurement some aggregate (e.g. average) of the measurement upon individual units of that sample?
- **ICQ-6:** Or, is the value of the measurement a measure of whether or not individual units of the sample passed or failed a certain threshold (e.g., does the sample in well 34 show up-regulation)?
- **ICQ-7:** What ought to be the measured value; that is, what is the expected value for that physical characteristic?
- **ICQ-8:** What are the tolerance specifications for a physical characteristic that is measured?
- **ICQ-9:** What is the unit of measurement for a physical characteristic of an entity?

Measurement Activities: In order to elaborate C-2, the following questions about measurement and inspection can be asked.

- **ICQ-10:** Is this an activity that performs measurement?
- **ICQ-11:** Is this an inspection activity?

Example 1.

Core-1 $holds(f,s)$ *Fluent f is true (i.e., holds) in situation s*
Core-2 $occurs_T(f,T)$ *Fluent f occurs in time period T*

Example 2.

Core-3 organization_agent(Oa)   *Oa is an organization_agent*
      \<Oa\>   an individual or group of individuals
Core-4 agent_constraint(A,c($\underline{X}$))
$holds$(agent_constraint(Oa,c($\underline{X}$)),s) $\leftrightarrow$ Φ(Oa,$\underline{X}$,s).
       \<s\>   a given situation
       \<Oa\>   an organization agent which seeks to achieve a goal in situation s
       \<$\underline{X}$\>   entities that must be represented in order to represent the constraints on Oa; $\underline{X}$ is a vector with none,
            one, or more entities
       \<c($\underline{X}$)\> predicate name for the agent constraint
       \<Φ(Oa,$\underline{X}$,s)\> a first-order logic expression for the constraint described as c($\underline{X}$)
Core-5 organizational_constraint(C)
       \<C\>   unique identifier for each instance of c($\underline{X}$)

Measurement Points: In order to elaborate C-3, the elemental piece of information needed to make a quality assessment can be represented as the value of a measurement taken at a point in time. Following are questions about quality that build on this.

- **ICQ-12:** What is the measured value for a physical characteristic at a given *point* in time?
- **ICQ-13:** What are the measured values for a physical characteristic during a given *period* of time?
- **ICQ-14:** Is an entity of "good" quality at a given point in time?
- **ICQ-15:** Is an entity of "bad" quality at a given point in time?
- **ICQ-16:** Is an entity of conforming quality over a given period of time?

The terminology with which these questions can be more formally posed is developed next.

Terminology & Formal Competency Questions

TOVE Core Ontologies: TOVE measurement ontology terms are defined with propositions (or Boolean terms) from the TOVE Core Ontologies, which are founded upon a first-order language for representing dynamically changing worlds. This first-order language is called the situation calculus (McCarthy & Hayes, 1969). In situation calculus, each perturbation to the modeled world changes the world from one *situation* \<s\> to another. If the truth value of a term, which describes an entity in this world or a relationship between entities, varies from situation to situation, then the term is a *fluent* \<f\>. A fluent *holds* in a given situation, if the term is true in that situation. A fluent *occurs in a given time period* \<T\>, if for all situations which occur during that period, the fluent holds. Note that unless otherwise stated, all predicates of the ontology are fluents, e.g., the predicate presented as organization_agent(Oa)

Example 3.

Core-6	resource(R)	*R is a resource*
Core-7	ru(Rt)	*Rt is a traceable resource unit*
Core-8	has_tru(R,Rt)	*Rt is an individual unit within R*
Core-9	activity(A)	*A is an activity*
Core-10	primitive_activity(A)	*A is a primitive activity*
Core-11	has_subactivity(A,Ao)	*Ao is a subactivity in activity A*
Core-12	consume_res_tru(A,Rt); consume_res_tru(A,R)	
Core-13	produce_res_tru(A,Rt); produce_res_tru(A,R)	
Core-14	use_res_tru(A,Rt); use_res_tru(A,R)	

Example 4.

Core-15	activity_duration(A,T)	*Activity A is performed within time T*
Core-16	has_point(T,Tp)	*Tp is a time point in T*

Example 5.

Core-17	has_subclass(X,X_o)	*Class X has a subclass Xo*
Core-18	has_attribute(X,Ch)	*X has an attribute Ch*
Core-19	has_attribute_value(X,Ch,V)	*Attribute Ch of Object X has value V*
	<X> an object	
	$<X_o>$ a subclass of X	
	<Ch> an attribute of X	
	<V> value of Atr for X	

Example 6.

Term-1	quality_requirement(Qr)	Qr is a quality requirement
Term-2	measured_attribute(At)	At is a measured attribute
Term-3	samples_attribute(Ch,At)	At is a measured attribute sample of Ch.

Example 7.

Term-4	has_sample_sizing(At,Sz)	*Measured Attribute At has sample size Sz*
Term-5	has_sampling_plan(At,Sp)	*At has type of sampling plan Sp*

is actually expressed in the ontology as holds(organization_agent(Oa),s). Also, a variable or parameter of a term is denoted within <> brackets when the term is defined. <R> denotes that 'R' is one of the variables of the term *resource*, e.g. if R='chair' for the expression resource(R) then this is read as 'a chair is a resource.' See Example 1.

An *agent constraint* is a special fluent representing a constraint upon an *organization agent* that must be satisfied in order for that agent to achieve some goal. See Example 2.

Some agent constraints refer to use of *resource* <R>—a conceptual or prototypical entity such as arm assembly—and *tru* (traceable resource unit) <Rt>—a physical set of individual units of a resource, such as 'lot#123 of arm assemblies' (Kim, Fox, & Grüninger, 1999), or a microarray assay plate or reagents. An *activity* <A> may *consume trus* of one resource to *produce trus*

Example 8.

Term-6	has_standard_value(At,Mu)	*Mu is a standard value of At*
Term-7	has_specification_set(At,SL)	*SL is a specification set of At*
Term-8	has_unit_of_measurement(At,U)	*At is measured using unit U*

Figure 3. Measurement description system data model
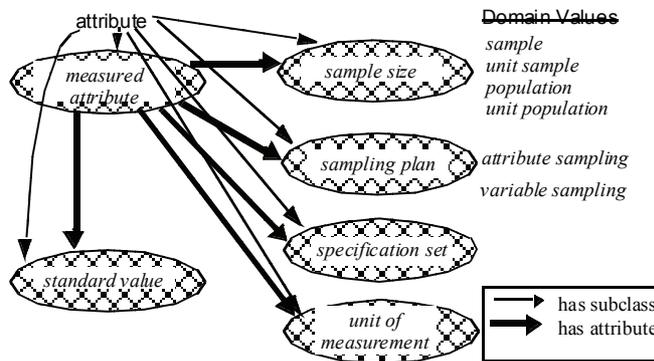
Example 9.

Term-9	measuring_resource(R)
Term-10	primitive_measure(A)
Term-11	measure(A)
Term-12	inspect_and_test(A)

of another resource, and may *use trus* of yet another resource; if a tru is used, individual units of the resource that comprise it are not consumed so they are available for use after an activity's execution. Activities can be hierarchically modeled as comprised of *subactivities* <Ao>; a *primitive activity* has no sub-activities. **Core-12** can be interpreted as "consume resource or tru." This interpretation also holds for "produce" and "use." See Example 3.

The time duration <T> within which an activity <A> is performed can be represented as a sequential list of discrete time points <Tp>. See Example 4.

Finally, general object-oriented constructs are represented as seen in Example 5.

Measurement Description System: A *quality requirement* <Qr> specifies physical characteristics <Ch> of a resource—e.g., "the number of genes examined or the number of probes used per gene"—that has a bearing on quality, as well as constraints that define acceptable quality. A *measured attribute* <At> of a sample of a tru—e.g.*,* "average light intensity and color"—is the explicit, exact descriptor of what is measured. So, the following are used to formally express ICQ-1 to ICQ-3. See Example 6.

Example 10.

Term-13	measurement_pt(Rt,At,Mp,Tp)	*tru Rt of Attribute At measured using measurement point Mp at time point Tp*
Term-14	conformance_pt(Q,Rt,At,Tp)	*Q is a conformance/non-conformance point*
Term-15	nonconformance_pt(Q,Rt,At,Tp)	*with related tru Rt, attribute At at time Tp.*

Example 11.

Term-16   conforming_quality(X,Qr)

Example 12.

CQ-1   Does there exist a quality requirement '$\theta\rho$'[1] in a situation 'σ'?
holds(quality_requirement('$\theta\rho$'),σ).

CQ-2   Does there exist a measured attribute <At> for a tru 'κ' in a situation 'σ'?
$\exists$At [*holds*(tru('κ'),σ) $\wedge$ *holds*(has_attribute('κ',At),σ) $\wedge$ *holds*(measured_attribute(At),σ)].
$\exists$At [*holds*(tru('κ') $\wedge$ has_attribute('κ',At)$\wedge$measured_attribute(At),σ)].

CQ-4   For a measured attribute 'α' of a tru 'κ' in a given situation 'σ', does it have a 'unit_sample' or 'sample' sample sizing plan?
holds(tru(κ),σ) $\wedge$ *holds*(has_attribute(κ,α),σ) $\wedge$ *holds*(measured_attribute(α),σ) $\wedge$
(*holds*(has_sample_sizing(α,unit_sample),σ) $\vee$ *holds*(has_sample_sizing(α,sample),σ)).

CQ-8   For a measured attribute 'α' in a given situation 'σ', does it have a specification set (expressed as an interval set [<T_1>,<T_2>] or a list [<W_i>]?
$\exists T_1 \exists T_2 \exists \{W_i\}$ [*holds*(measured_attribute('α'),σ) $\wedge$
(*holds*(has_specification_set('α',[T_1,T_2]),σ) $\vee$ *holds*(has_specification_set(α,$\{W_i\}$),σ))].

There are two additional issues regarding sampling:

- **Sample Size:** How many individuals in a set are measured to model the characteristics of the set?
- **Sampling Plan:** When determining an aggregate value from the sample, does it refer directly to the actual attribute that is physically measured or is the reference indirect?

Therefore, the terms in Example 7 are used to express ICQ-4 to ICQ-6.

To express ICQ-7 to ICQ-9, a *standard value* μ <Mu> for what the value of a measured attribute <At> ought to be is represented, as well as a function of μ and σ^2 ($f(\mu,\sigma^2)$) and an operator ($\otimes$). *Unit of measurement* <U> is also represented. For a measured attribute <At>, a subset [$\mu - k\sigma^2$, $\mu + k\sigma^2$], a generic term called a *specification set* <SL> is represented, where elements of this subset denote acceptable measurement values. See Example 8.

Measurement Activities: The simplest measurement action is the measurement of one measured attribute of one tru at one point in time. When this measurement is performed using a special resource <R> called a *measuring resource*, this activity <A> is a *primitive measure* activity. A primitive measure activity or an aggregation of primitive measure activities is a *measure* activity. An *inspection and test* activity is a form of a measure activity. The following terms then are used to formally express

Example 13.

Cons-1 A measured attribute must be an attribute of a tru.
Cons-2 measured attribute must be sampled from an attribute of a resource
$\forall At \forall s$ [*holds*(measured_attribute(At),s) $\rightarrow$
  $\exists Atr \exists R$ (*holds*(samples_attribute(Atr,At),s) $\wedge$
  holds(has_attribute(R,Atr),s) $\wedge$ *holds*(resource(R),s))].
  <At> measured attribute
  <Atr> attribute sampled for At
  <Rt> tru for which At is an attribute
  <R> resource for which Atr is an attribute
  <s> an extant or hypothetical situation

Example 14.

Cons-3 All measured attributes must have a specification set, and the standard value for that measured attribute
  must be an element of the specification set.
$\forall At \forall Mu \forall s$ [*holds*(has_standard_value(At,Mu),s) $\rightarrow$
  $\exists T_1 \exists T_2$(*holds*(has_specification_set(At,[T_1,T_2]),s) $\wedge$ $T_1 \leq Mu \leq T_2$) $\vee$
  $\exists \{W_i\}$ (*holds*(has_specification_set(At,\{W_i\}),s) $\wedge$ $Mu \in \{W_i\}$)]
  <At> a measured attribute
  [<T_1>,<T_2>] upper and lower bounds of a specification set for a measured attribute of ratio scale
  \{W_i\} a set of "acceptable" values for the measured attribute
  <Mu> the standard value for At
  <s> an extant or hypothetical situation

ICQ-10 and ICQ-11. See Example 9.

Measurement Point: To formally express ICQ-12 and ICQ-13, the result of a measurement activity is represented using a *measurement point* <Mp>, relating the value of the measurement, and concomitantly the measured attribute <At>, the time of measurement <Tp>, and tru measured <Rt>. See Example 10.

The following term then represents the concept that "quality is conformance to requirements." See Example 11.

Using these terms, ICQ-14 to ICQ-16 are expressed. Informal competency questions are stated formally next.

Formal Competency Questions

For brevity, only the competency questions for the measurement description system are presented. The informal competency question is re-stated in English using the terminology developed from the ontology, then expressed formally in First-Order Logic.

Measurement Description System: See Example 12.

Axioms

The axioms in the ontology are presented next. Once again, we only describe the axioms for the measurement description system for lack of space.

Measurement Description System: A categorical characterization of the content of all quality requirements is difficult; axioms about quality requirements true for all enterprises in all industries are not likely to exist. Rather, quality requirement is stated as a *primitive term (*PT*)*—a term that is instantiated and stated as a fact in the populated enterprise model—in the

Table 1. Steps for using an advisor for demonstrating competency

Step #	**Bioinformatician View**: Using the Advisor to Analyze Specific Enterprise		**Ontology Builder View**: Using the Advisor to Evaluate Competency of Ontology
1	Stating facts about experimental conditions	⇔	Representing populated enterprise model
2	Stating explicit gene-gene relationships	⇔	Representing formal competency questions
3	Stating data dictionary of experimental terms	⇔	Representing ontology terminology and axioms
4	Answering queries	⇔	Deducing answers to formal competency questions

⇔ denotes translation between knowledge about an enterprise and how that knowledge is represented using an ontology

TOVE measurement ontology. That is, the enterprise modeler determines that a given agent constraint is a quality requirement and states this as a fact. All *definition* (Defn) axioms are ultimately formally defined in terms of primitive terms. By applying *constraint* (**Cons**) axioms, proper use of primitive terms is enforced. Primitive terms are populated (instantiated) as *ground terms*; e.g. a fact that 'widget' is a resource is represented as a ground term, resource (widget), which is an instance of the primitive term, resource(R). Measured attribute and samples attribute are also primitive terms.

The quality of an activity is evaluated by the quality of resources associated with that activity; and the quality of a resource (prototypical product) is gauged by the quality of trus comprised of individual units of that resource. The constraint axioms in Example 13 express this.

These axioms ensure valid answers for ICQ-2. Additional such constraints constrain the use of the primitive terms has_sample_sizing, has_sampling_plan, has_standard_value, has_specification_set, and has_unit_of_measurement. The axiom in Example 14 ensures valid answers for ICQ-8.

DEMONSTRATION OF COMPETENCY

A demonstration of competency entails using the TOVE ontological engineering methodology for the following:

- **Microarray Analysis:** Does an ontology support problem-solving tasks for a specific pathway—wing development in *Drosophila melanogaster*?
- **Ontology Development:** Are ontology competency questions—which by design do not refer to any specific domain—answered, thus demonstrating that the ontology can be used to support similar problem-solving tasks for other domains?

The first question characterizes requirements for an analyst familiar with the domain, not the ontology's language or representations; the second question is relevant for an engineer with expertise in ontology-based enterprise modeling who is unfamiliar with details of the specific domain modeled. An *advisor* is a user interface and access routines front-end software to an ontology-based enterprise model. Table 1 lists how an advisor answers the two different questions.

Table 2.

#1	Stating facts about the process	⇔	Representing populated model
Emerging	"collections are fluorescently labeled"	⇔	*holds(*activity(process_fl_label), sw_actual)
Traditional	"There are many activities…" from Motivating Scenario	⇔	*holds*(activity(process_wp_hcpf_260_1） , sv_actual).

#2	Stating queries for analyzing process	⇔	Representing formal competency questions
Emerging	"Is this an activity in which a specific gene is determined to be under-expressed?"	⇔	*holds(*inspect_and_test(α),σ).
Traditional	"Is this an activity in which products are determined to be non-prime?"	⇔	*holds*(inspect_and_test(α),σ).

#3	Stating data dictionary of actual process's terms	⇔	Representing ontology terminology and axioms
Emerging	"An activity where a gene is determined to be unexpressed is an inspect and test activity according to the TOVE Measurement Ontology"	⇔	*holds(measure(inspect_and_test_exp_gene1), sv_actual)*
Traditional	"An activity where products are determined to be non-prime is an inspect and test activity according to the TOVE Measurement Ontology."	⇔	*holds*(measure(inspect_and_test_wp_hcpf_1),sv_actual).

#4	Answering queries	⇔	Deducing answers to formal competency questions
Emerging	"The inspections are done through an activity which measures the average light intensity for a specific cell using a light sensor"	⇔	*holds(*inspect_and_test(C_{ij}), s)
Traditional	"The inspections are done through an activity which measures the average coil length using a sensor."	⇔	*holds*(inspect_and_test(A),s) $\equiv$ …

We now provide the actual process in which each of the steps in Table 1 manifests in the ontology by identifying parts of the motivating scenario that correspond to the ontology predicates (see Table 2). This can be demonstrated computationally using the ISO 9000 Quality Advisor (Kim & Fox, 2002), a tool for evaluating ISO 9000 compliance of an enterprise modeled using ontologies. Evaluation entails answering questions about an enterprise's inspection and measurement system, so it can be used for enterprise analysis, as well as for assessing the competency of the measurement ontology.

A complete demonstration of competency using all of the components of the ontology is beyond the scope of this article because of length restrictions. A full demonstration is shown in (Kim, 1999), which also includes a demonstration of generalizability of the measurement ontology.

Table 3.

Manufacturing (Basis for Measurement Ontology)	Measuring QoS for Web services	Tracking cattle via RFID
ICQ-2: What is the physical characteristic that is measured?	What is the QoS metric that is measured? Answer could be mean response time, and facts could be represented using the ontology as has_attribute(message121,response_time) measured_attribute(mean_response_time)	Which data about beef quality needs to be collected? Answers could be fat thickness, and facts could be represented using the ontology as has_attribute(cowcarcass1871,fat_thickness).
ICQ-3: Is every entity that is produced measured?	Is the response time for every message measured? Answer would be yes and represented as Has_sample_sizing(mean_response_time,sample_population) to signify that the population is sampled—i.e. response times for all packets that comprise a message are calculated. If only some packets were sampled then this would be represented as has_sample_sizing(mean_response_time,sample).	Is every carcass measured? Answer would be yes and represented as has_sample_sizing(fat_thickness,unit_sample) to signify that each unit is sampled.
ICQ-5: If a sample is taken and measured, is the value for the measurement some aggregate (e.g. average) of the measurement upon individual units of that sample?	Are the response times for a message averaged? Answer is yes, since the mean response time for a message is the average of the response times of all packets that comprise that message. This would be represented as samples_attribute(mean_response_time,response_time) and has_sampling_plan(mean_response_time,attribute_sampling).	Is the mean fat thickness of each carcass from a lot averaged? Answer is no, since fat thickness is tracked individually and not associated with the lots that identify the herd of origin.

COMMENTARY ON GENERALIZABILITY

In our full demonstration of generalizability (Kim, 1999), we show that the measurement ontology's representations could be used to perform similar tasks as a related application. Specifically, it is shown that parts of SAP's quality control management model (SAP-AG, 1995) could be reasonably "reduced" or represented using only ontology representations. The implication then is that a query answerable using the SAP™ data model is answerable as a competency question posed and answered using the ontology. That is, the ontology is as "competent" as SAP™ with respect to the reduced set of competency questions.

Whereas that demonstration provides evidence of intra-domain, inter-application (using different models [or ontologies] in a similar domain) generalizability of the ontology, what we have shown in Table 2 provides evidence of inter-domain, intra-application (using same model [ontology] for different domains) generalizability. That is, we show that same set of general measurement representations can be used to pose and answer competency questions from two different domains.

We cannot claim that the measurement ontology can be a "plug-and-play" ontology for all other domains. Rather, we can claim that the ontology's representations can be used to define terms and express axioms for more domain-specific ontologies. The

basis of this claim is the core of the methodology used: competency questions. If informal competency questions associated with a specific domain can be reasonably posed as formal competency questions represented using measurement ontology representations, we can make an inductive argument that the measurement ontology is generalizable to that domain. To that end, we show in Table 3 how the informal competency questions from measuring the quality of service (QoS) of Web services and agriculture (tracking cattle) domains relate to the informal competency questions that motivated this ontology. It is evident then that the measurement ontology can serve as a very re-usable, generalizable basis to develop more committed ontologies of highly diverse fields, from measuring QoS to tracking cattle.

CONCLUDING REMARKS

There is clearly a need for an ontology that represents both the obvious and more subtle fundamental concepts in measurement. This is certainly true in emerging domains like bioinformatics where quality plays such a vital and critical role in discovery. A measurement ontology is particularly useful for builders of Semantic Web ontologies of emerging domains because of the importance of measurement in these domains, which are firmly grounded in the physical sciences. The measurement concepts represented in this article are summarized as follows:

- A system for assessing measurements includes the appropriate *measured attribute*, as well as its *standard value* (μ), *sampling plan* and *size*, *specification set* of "acceptable values" of $f(\mu, \sigma^2)$, and *unit of measurement*. Measurements of attributes are recorded as *measurement points* in time that are assigned a value as a result of some *measurement activity*. These representations are the basic ones necessary to model any form of measurement.

- Quality can be represented as some composition of *conformance points*, which are "conforming" measurement points with respect to some *quality requirement*. Representing quality requirements, measurement points, and conformance points makes it possible to model and assess any entity within an enterprise as of *conforming quality*.

These concepts are formally represented by posing competency questions, analyzing the domain of measurement, stating assumptions, and developing terminology and axioms. Then, the competency of the ontology and its capability for use to gain insights about an enterprise are demonstrated by automatically deducing answers to questions such as:

- **Quality Assessment System Competency Questions:** What is the standard value for a given measured attribute of an entity? What is its specification set? How is the measured attribute sampled?

- **Measurement and Conformance Points Competency Questions:** What is the measurement point for a measured attribute? Is it a conformance point? Over a period of time, is an entity of conforming quality?

Measurement theory is more than just measurement unit conversions and quantities. The root of our contribution is

that we present an ontology that represents more concepts than other ontologies. This general ontology is then a useful reference regardless of whether it serves as a building block for traditional or emerging domains, or for the Semantic Web or closed networks. With such an ontology in tow, developers of Semantic Web ontologies of emerging domains as diverse as biology and geography have a reference for ensuring that important and subtle measurement concepts are diligently represented in their ontologies and not ignored.

The clear future direction for this work is to (1) transform this ontology so that its representations are more general, and that ontological commitment made towards enterprise applications is further abstracted out; (2) develop separate domain specific ontologies of emerging domains that actually define and restrict their terms using the general ontology developed in (1) (as opposed to the realistic use of this ontology, which is to serve as inspiration and reference); and (3) implement both the general and the domain specific ontologies in the *de facto* ontology language of the Semantic Web, OWL.

Overall we believe that we have inspired builders of emerging domain ontologies to incorporate important commonsense of their domains—that of obvious and subtle measurement concepts—in their ontologies, so that their ontologies will be more robust and sharable for functioning on the Semantic Web.

REFERENCES

Allen, J. F. (1983). Maintaining knowledge about temporal intervals. *Communications of the ACM, 26*(11), 832-843.

Ashburner, M., Ball, C., Blake, J., Botstein, D., Butler, H., Cherry, J., et al. (2000). Gene ontology: Tool for the unification of biology. *National Genetics, 25*, 25-29.

Attie, A. D. (2003). The new industrialized approach to biology. *Cell Biology Education, 2*, 150-151.

Bechhofer, S., Goble, C., & Horrocks, I. (2001, July 30-August 1). *DAML+OIL Is not Enough.* Paper presented at the First Semantic Web Working Symposium (SWWS-01), Stanford, CA.

Berners-Lee, T., Hendler, J., & Lassila, O. (2001). The Semantic Web. *Scientific American, 284*(5), 34-43.

Bowers, S., & Ludäscher, B. (2004). *An ontology-driven framework for data transformation in scientific workflows.* Paper presented at the International Workshop on Data Integration in the Life Sciences (DILS'04), Leipzig, Germany.

Brilhante, V., & Robertson, D. (2001). Metadata-Supported Automated Ecological Modelling. In C. Rautenstrauch (Ed.), *Environmental information systems in industry and public administration*. Hershey, PA: Idea Group Publishing.

Campbell, A. E., & Shapiro, S. C. (1995). *Ontological mediation: An overview.* Paper presented at the IJCAI Workshop on Basic Ontological Issues in Knowledge Sharing.

Chen, Y., Zhou, L., & Zhang, D. (2006). Ontology-supported Web service composition: An approach to service-oriented knowledge management in corporate services. *Journal of Database Management, 17*(1), 67-84.

Davies, J., Duke, A., & Stonkus, A. (2002). *OntoShare: Using ontologies for knowledge sharing.* Paper presented at the Semantic Web Workshop of the

WWW Conference, Hawaii, HI.

Duggan, D. J., Chen, M., Meltzer, P., & Trent, J. (1999). Expression profiling using cDNA microarrays. *Nature Genetics, 21*, 10-14.

Fenton, N. E. (1994). Software measurement: a necessary scientific basis. *IEEE Transactions on Software Engineering, 20*(3), 199-206.

Fikes, R., & Farquhar, A. (1999). Distributed repositories of highly expressive reusable ontologies. *IEEE Intelligent Systems, 14*(2), 73-79.

Finkelstein, L. (1984). A review of the fundamental concepts of measurement. *Measurement, 2*(1), 25-34.

Fox, M. S. (1992). The TOVE project, towards a common sense model of the enterprise. In C. Petrie (Ed.), *Enterprise integration*. Cambridge, MA: MIT Press.

Fox, M. S., & Gruninger, M. (1998). Enterprise modelling. *AI Magazine, 19*(3), 109-121.

Gandon, F. (2001, October 22-24). *Engineering an ontology for a multi-agents corporate memory system.* Paper presented at the ISMICK 2001 Eighth International Symposium on the Management of Industrial and Corporate Knowledge, Université de Technologie de Compiègne, France.

Gandon, F. L., & Sadeh, N. M. (2004). Semantic Web technologies to reconcile privacy and context awareness. *Journal of Web Semantics, 1*(3), 241-260.

Goh, C. H., Bressan, S., Madnick, S., & Siegel, M. (1999). Context interchange: New features and formalisms for the intelligent integration of information. *ACM Transactions on Information Systems, 17*(3), 270-293.

Grady, J. O. (1993). *System requirements analysis*: McGraw-Hill Inc.

Gruber, T. R. (1993, March). *Towards principles for the design of pntologies used for knowledge sharing.* Paper presented at the International Workshop on Formal Ontology, Padova, Italy.

Gruber, T. R., & Olsen, G. R. (1994). *An ontology for engineering mathematics.* Paper presented at the Fourth International Conference on Principles of Knowledge Representation and Reasoning, Bonn, Germany.

Gruninger, M., & Fox, M. S. (1995). *Methodology for the design and evaluation of ontologies.* Paper presented at the Workshop on Basic Ontological Issues in Knowledge Sharing, IJCAI-95, Montreal, Canada.

Hayes, P. J. (1985). Naive physics I: Ontology for liquids. In J. Hobbs & B. Moore (Eds.), *Theories of the commonsense world* (pp. 71-89). Ablex Publishing Corp.

Heflin, J. (2000, April 3, 2000). Measurement ontology 1.0 (draft). Retrieved August 15, 2004, from http://www.cs.umd.edu/projects/plus/SHOE/onts/measure1.0.html

Hewett, M., Oliver, D. E., Rubin, D. L., Easton, K. L., Stuart, J. M., Altman, R. B*., et al.* (2002). PharmGKB: the pharmaco-genetics knowledge base. *Nucleic Acids Research, 30*(1), 163-165.

Khatri, V., Ram, S., & Snodgrass, R. T. (2004). Augmenting a conceptual model with geospatiotemporal annotations. *IEEE Transactions on Knowledge and Data Engineering, 16*(11), 1324-1338.

Kim, H. M. (1999). *Representing and reasoning about quality using enterprise*

models. Unpublished PhD thesis, Department of Industrial Engineering, University of Toronto, Toronto, Ontario, Canada.

Kim, H. M. (2002, January 7-10). *XML-hoo!: A prototype application for intelligent query of XML documents using domain-specific ontologies*. Paper presented at the 35th Annual Hawaii International Conference on Systems Science (HICSS-35), Hawaii, HI.

Kim, H. M., & Fox, M. S. (2002, January 4-7). *Using enterprise reference models for automated ISO 9000 compliance evaluation.* Paper presented at the 35th Hawaii International Conference on Systems Science (HICSS), Big Island, HI.

Kim, H. M., Fox, M. S., & Grüninger, M. (1999). An ontology for quality management: Enabling quality problem identification and tracing. *BT Technology Journal, 17*(4), 131-139.

Klischewski, A. R., & Jeenicke, M. (2004). *Semantic Web technologies for information management within e-government services*. Paper presented at the 37th Annual Hawaii International Conference on System Sciences (HICSS'04), Big Island, HI.

Kuipers, B. J. (1986). Qualitative simulation. *Artificial Intelligence, 29*(3), 289-338.

Lenat, D. B. (1995). CYC: A large-scale investment in knowledge infrastructure. *Communications of the ACM, 38*(11), 33-38.

Lesk, A. M. (2005). *Databasse annotation in molecular biology, principles and practice*: John Wiley & Sons Ltd.

Lozano-Tello, A., & Gomez-Perez, A. (2004). ONTOMETRIC: A method to choose the appropriate ontology. *Journal of Database Management, 15*(2), 1-18.

Mars, N. J. I. (1993, March 10-13). *An ontology of measurement units*. Paper presented at the International Workshop on Formal Ontologies in Conceptual Analysis and Knowledge Representation, Padova, Italy.

Massey, B. S. (1971). *Units, dimensional analysis, and physical similarity*. London: Van Nostrom Reinhold.

McCarthy, J., & Hayes, P. J. (1969). Some philosophical problems from the standpoint of AI. In B. Meltzer & D. Michie (Eds.), *Machine intelligence* (Vol. 4, pp. 463-501). Edinburgh, UK: Edinburgh University Press.

McGuinness, D. L., & van Harmelen, F. (2003). *OWL Web Ontology Language overview* (No. CR-owl-features-20030818). W3C.

Milton, S. K., & Kazmierczak, E. (2004). An ontology of data modelling languages: A study using a common-sense realistic ontology. *Journal of Database Management, 15*(2), 19-38.

Mount, D. W. (2004). *Bioinformatics sequence and genome analysis*. Cold Spring Harbor, New York: Cold Spring Harbor Laboratory Press.

Narens, L. (1985). *Abstract measurement theory*. Cambridge, MA: MIT Press.

Novak, G. S., Jr. (1995). Conversion of units of measurement. *IEEE Transactions on Software Engineering, 21*(8), 651-661.

Noy, N. F., & Hafner, C. D. (1997). The state of the art in ontology sesign: A survey and comparative review. *AI Magazine, 18*(3), 53-74.

Nozick, L. K., Turnquist, M. A., Jones, D. A., Davis, J. R., & Lawton, C. R. (2004). *Assessing the Performance*

of Interdependent Infrastructures and Optimizing Investments. Paper presented at the 37th Annual Hawaii International Conference on System Sciences, Big Island, HI.

Pease, A., & Niles, I. (2002). IEEE standard upper ontology: A progress report. *Knowledge Engineering Review, 17*, 65-70.

Ram, S., & Wei, W. (2004). *Modeling the semantics of 3D protein structures.* Paper presented at the ER 2004, Shanghai, China.

Reiger, C., & Grinberg, M. (1977). *The declarative representation and procedural simulations of causality in physical mechanisms.* Paper presented at the Joint Conference on Artificial Intelligence.

Retz-Schmidt, G. (1988). Various views on spatial prepositions. *AI Magazine, 9*(2), 95-105.

Roberts, F. (1979). *Measurement theory with applications to decision making, utility and the social sciences.* Reading, MA: Addison-Wesley.

Ronco, L., Grossel, M., Zimmer, M., & Socash, T. (2003). *Modules in emerging fields. Vol 4: Genomics and proteomics.*

SAP-AG. (1995). *SAP R/3 system: Quality management* (No. 4.6). Neurottstrasse 16, 69190 Walldorf, Germany: SAP AG.

Scheaffer, R. L., & McClave, J. T. (1982). *Statistics for engineers*. Boston, MA: PWS Publishers.

Schlenoff, C., Balakirsky, S., Uschold, M., Provine, R., & Smith, S. (2003). Using ontologies to aid navigation planning in autonomous vehicles. *The knowledge engineering review, 18*, 243-255.

Stevens, R., Goble, C., Horrocks, I., & Bechhofer, S. (2002). Building a bioinformatics ontology using OIL. *IEEE Transactions on Information Technology in Biomedicine, 6*(2), 135-141.

Stevens, R., Goble, C. A., & Bechhofer, S. (2000). Ontology-based knowledge representation for bioinformatics. *Briefings in Bioinformatics, 1*(4), 398-414.

Stoeckert Jr., C. J., & Parkinson, H. (2004). The MGED ontology: A framework for describing functional genomics experiments. *Comparative and Functional Genomics, 4*(1), 127-132.

Uschold, M., & Gruninger, M. (1996). Ontologies: Principles, methods and applications. *Knowledge Engineering Review, 11*(2), 93-136.

Wroe, C., Stevens, R., Goble, C., & Ashburner, M. (2003). A methodology to migrate the gene ontology to a description logic environment using DAML+OIL. *Pacific Symposium on Biocomputing, 8*, 624-635.

ENDNOTE

[1] Facts or constants (as opposed to variables) expressed in competency questions are denoted in Greek letters within single quotes.

Henry M. Kim is an associate professor of information systems at the Schulich School of Business, York University in Toronto, Canada. He is interested in information systems that facilitate data and knowledge for enterprise modeling, knowledge management, and e-commerce. Dr. Kim's articles have appeared in journals such as the Communications of the ACM, Internet Research, BT Technology Journal, Business Process Management Journal, *and* Operational Research: An International Journal. *He has consulted to industries and researched in laboratories in Canada, U.S., UK, and Australia. He received a PhD in industrial engineering from the University of Toronto.*

Arijit Sengupta is an assistant professor of information systems and operations management at the Raj Soin College of Business at Wright State University, Dayton, Ohio. He received his PhD in computer science from Indiana University. Prior to joining Wright State, Dr. Sengupta served as faculty at Kelley School of Business at Indiana University and the Robinson College of Business at Georgia State University. Dr. Sengupta's research areas are in databases and XML, specifically in modeling, query languages, data mining, and human-computer interaction. He has published over 30 scholarly articles in leading journals and conferences, as well as authored several books and book chapters. He is a member of the Information Technology Committee of the Decision Sciences Institute and has led several of the information technology innovation projects for the institute. He has served in the program committee of several international conferences.

Mark S. Fox is a professor of industrial engineering with cross appointments in the Department of Computer Science and Faculty of Management Science at the University of Toronto. Dr. Fox is head of the Enterprise Integration Laboratory and holder of the NSERC Industrial Research Chair in Enterprise Integration. He is also co-founder, chairman and CEO of Novator Systems and co-founder and managing director of ChocolatePlanet.com. Prior to his return to Toronto, he was an associate professor of computer science and robotics and director of the Center for Integrated Manufacturing Systems of The Robotics Institute at Carnegie Mellon University. He is also a co-founder and past president of Carnegie Group Inc, a knowledge-based software company that focuses on engineering, manufacturing, and telecommunications applications. He received a BSc in computer science from the University of Toronto in 1975, and his PhD in computer science from Carnegie-Mellon University in 1983.

Mehmet M. Dalkilic is an assistant professor in the School of Informatics at Indiana University, Bloomington. He graduated from Indiana University with a PhD in computer science in June 2000. He is the curriculum coordinator for the introductory courses for undergraduates and beginning graduate students in bioinformatics. His primary research interests include data mining, software engineering in the life sciences, and bioinformatics. In data mining, he works on developing and application of novel classification techniques. Another project is the study and application of entropy to get a better understanding of the structure underlying the relational model. In bioinformatics he works on motif discovery using De Bruijn graphs. He is also the principle designer of a newly released package to manage and view protein family annotation information (www.catpa.org).