Kim, Henry M. and Fox, Mark S. (2002). "Using Enterprise Reference Models for Automated ISO 9000 Compliance Evaluation", Proceedings of the 35th Hawaii International Conference on Systems Science.

Using Enterprise Reference Models for Automated ISO 9000 Compliance Evaluation

Henry M. Kim[1] & Mark S. Fox[2]
Schulich School of Business, York University, 4700 Keele St., Toronto, Ontario Canada M3J 1P3[1]
Department of Mechanical and Industrial Engineering, University of Toronto, 5 King's College Road, Toronto, Ontario Canada M5S 3G92
hkim@schulich.yorku.ca[1], msf@mie.utoronto.ca[2]

Abstract

A computational enterprise model representing key facets of an organization can be an effective tool to consider when planning an enterprise information architecture. For example, a specific organization's quality management business processes and organizational structures can be represented using such a model, and then compared to a reference model of "good" processes and structures, such as the ISO 9000 standards. The specific and reference models can be represented using common entities, attributes, and relationships—comprising general schema or data model—which are then formally defined and constrained. These definitions and constraints can be used as inference rules applied to the models. Hence identification of differences between the models as quality problems can be automatically inferred, as can the analysis and correction of problems. In this paper, the TOVE ISO 9000 Micro-Theory is presented as a formal reference model of quality goodness. ISO 9000 requirements represented as inference rules in the micro-theory are applied to facts about an organization's quality management processes and structures, and conformance or nonconformance to requirements is automatically inferred. TOVE Ontologies for Quality Modeling are the common data and logical (formal definitions and constraints) models of the reference and specific organization's models. The example use of the micro-theory demonstrates enterprise model use for a pre-audit, which lowers the cost and time for improving quality through achieving ISO 9000 compliance. Since these enterprise models are constructed using ontologies, benefits of using ontologies such as model re-usability and sharability can be reaped.

1. Introduction

At the heart of information systems like ERP, CRM, and supply chain management applications that operate the enterprise are "computational representations of the structure, activities, processes, information, resources, people, behavior, goals, and constraints of a business, government, or other enterprise" [1]. An important function served by such applications is quality management; software applications for statistical quality control, quality problem identification and control, and quality audit preparation are some examples.

In particular, though audit preparation is supported by some applications, automatic audits generally are not. This means that enterprise model support for the most pertinent quality audit—one required for achieving ISO 9000 compliance—is limited. Achieving ISO 9000 compliance signifies to an organization's customers that it has met stringent international standards on how it manages quality. In certain manufacturing industries such as petro-chemicals, ISO 9000 achievement is used to gain an advantage over competitors. As such, an organization planning its information architecture, particularly a manufacturing one, should consider a tool to assist in compliance evaluation in planning IT support for quality management.

Automatic ISO 9000 compliance evaluation entails applying a general reference model of conditions or characteristics of an enterprise that conforms to ISO 9000 to a specific enterprise model's structures, activities, information, etc. for quality management. If the specific model satisfies the conditions of the reference model, then a stronger statement can be made that the enterprise conforms to a subset of ISO 9000 requirements represented in the ISO 9000 reference model, which can be objectively, automatically evaluated and do not rely upon subjective judgment of a specially trained auditor. One way to do this is by developing the reference and specific models using common underlying building block data models called ontologies, which "consist 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" [2]. Ontology-based enterprise models' use is appropriate because they are models of various facets of quality management within the organization (recall definition of enterprise model), and modeled formally (recall definition of ontology).

Consider a formal model as represented in a formal language with limited syntax and semantics. Initial propositions are stated, and axioms are defined in terms of these propositions and other axioms so defined. Since limited syntax and semantics restrict model interpretation, precise inference of propositions is possible by applying axioms to initial propositions. Then computer algorithms automate this inference. For ISO 9000 compliance evaluation, a proposition such as "This organization complies to ISO 9001 requirement 4.10 – Inspection and Testing." can be automatically inferred this way:

1. Facts required to demonstrate compliance are represented as initial propositions in a computational model of a specific enterprise.
2. Rules for compliance are represented in a separate model built using same formal building blocks (language, syntax, vocabulary, and axioms—i.e. the same ontologies) as the specific enterprise model.
3. By applying the rules to the facts, truth of the compliance evaluation proposition is inferred. This inference is automatically performed using computers.

Using this approach, an automated pre-audit can be performed for testing compliance to objective ISO 9000 requirements, before an expensive audit for subjective requirements takes place. The pre-audit reduces the cost and time for achieving ISO 9000 compliance. Moreover, since the ISO 9000 reference model is separate from the specific enterprise model, it can be applied to models of different enterprises, as long as all the models are constructed using the same ontologies. Hence, the reference model is re-usable. Even if other enterprise models are not based on the same ontologies, semi-automatic ISO 9000 compliance evaluation is still possible. Terms that are precisely defined and constrained can be interpreted according to the intentions of the model builder by other applications. Formal axioms that translate representations of "native" ontologies used for the ISO 9000 reference model to "foreign" ontologies used by other applications can be reasonably developed to re-create an ISO 9000 compliance evaluation model using "foreign" ontologies. Hence, the reference model is sharable. Not only can ontology-based enterprise model use for automatic ISO 9000 compliance evaluation save time and money for an audit for one enterprise, similar compliance evaluation systems may be developed more cheaply and quickly because the models used are re-usable and sharable. The research question addressed then is this: Can ontologies be used to develop enterprise models that are both 1) useful for ISO 9000 compliance evaluation, and 2) sharable for evaluation across enterprises and re-usable for developing other quality management applications.

In this paper, literature in enterprise modeling, ontologies, and ISO 9000 quality is reviewed, and opportunities for ontology-based enterprise model use for ISO 9000 compliance evaluation are highlighted. Then, the methodology is presented, followed by the models. A prototype demonstration of the compliance evaluation software is presented. Finally, concluding remarks and future work are stated.

2. Literature Review

ISO 9000 is the collective name for three international quality assurance standards, definitions for standardized quality vocabulary, and various accompanying documents [3]. ISO 9000 certification means that an expert, independent auditor verifies that an organization's quality system complies with one of the ISO 9000 standards—ISO 9001, 9002, or 9003. ISO 9003 applies for organizations that only warehouse; ISO 9002 applies for organizations that also produce; and ISO 9001 applies for organizations that in addition perform design and development.

ISO 9000 software "range from products specifically designed to meet the demands of ISO 9000, such as data control and corrective and preventive actions, to more far-reaching titles that cover areas important to any quality system, such as statistical process control and failure analysis [4]. Most software tools that assist directly in ISO 9000 audits only provide checklists; questions posed to users are almost verbatim sentence-by-sentence dissections of ISO 9000 requirements. One that off-loads some objective audit decisions onto a computer is The Strategic Analyst™ [5], an expert system for internal, informal, ISO 9000 audit. It offers some 500 questions, where questions asked of the user depend on answers given to previous questions. It still employs a checklist approach, since its key input source is the user, not a model of the enterprise to be analyzed. As a result, information unknown to the user but represented elsewhere in the enterprise cannot be integrated for the audit.

Evaluating ISO 9000 compliance requires integrating information from different parts of the organization. At the heart of computer support for this enterprise integration—e.g. ERP—are enterprise models. For instance, ERP software mySAP™ [6] provides a quality module used to centralize quality functions such as document and engineering configuration controls. All modules use a common enterprise model, so it is possible, for example, to automatically trigger an inspection when goods are received. Customizing a general reference model pre-disposes an enterprise to construct their business processes a certain way. In this vein, strict use of the quality module pre-disposes practices that will make ISO 9000 compliance easier. However, there is no module for ISO 9000 compliance auditing for mySAP™ nor other ERP software such as iBaan™ [7]. Following is a rationale for this: Whereas ERP systems use enterprise models primarily for operations of organizations, model use for ISO 9000 compliance evaluation is better classified as for organizational analysis and design—e.g. for business process re-engineering (BPR) [8] and enterprise engineering [9].

For analysis and design, prescriptions such as ISO 9000 compliance requirements and BPR heuristics must be applied to the enterprise model. Query models of analysis such as RDBMS based or expert systems represent prescriptions as queries whose answers reveal insights about the modeled enterprise. Queries expressed declaratively (e.g. for expert systems) rather than procedurally (e.g. in SQL) represent prescriptions de-coupled from algorithms or procedures needed for query answer. Declarative queries are answered using deduction, which is supported only if queries are formally expressed. Because of their data-application independence and precision due to formality, formal declarative models are re-usable and sharable—important characteristics if the ISO 9000 compliance of different organizations is to be evaluated.

The SCOPE project [10] develops a formal, declarative, query enterprise model of analysis for software quality. “Products” (programs, specifications, requirements, and documentation) and quality metrics of software engineering are represented in an object model; the processes are represented as declarative rules in a semantic model, which dictate transition from one object model state to another. Query answers are deduced by applying rules to an instantiated object model. Applicable solely for software production, only a few of these rules mention ISO 9000. Nevertheless, it is an example of enterprise model use for automatic ISO 9000 compliance evaluation, and therefore can be built upon.

To summarize, effective ISO 9000 compliance evaluation requires capability to integrate information from different parts of the organization. Enterprise models enable integrated decision-making. Specifically model use for analysis and design, rather than enterprise operations, is appropriate for compliance evaluation. For re-use and sharing, it is desirable that models answer queries expressed declaratively and formally. Ontology-based enterprise models do exactly this; re-use of knowledge bases is the main rationale for ontology development [11].

Ontology development efforts can be classified into standards for sharing heterogeneous ontologies such as OIL [12] and Ontolingua [13]; natural language processing such as μKosmos [14]; and task performance such as real-world common sense reasoning [15] logistics planning [16] and enterprise modeling [17].

Extensive ontologies for enterprise modeling are also developed by the TOVE project. Ontologies of activity-state, causality, time, resource, and organization structure fundamental to describe any enterprise are collectively called the Core Ontologies [18], and used as building blocks to construct additional ontologies peripheral to the core, such as the Cost [19] Ontologies and Ontologies for Quality Modeling [20].

If Quality Ontologies are developed using Core Ontologies, it is reasonable that ontologies for ISO 9000 compliance be developed using Quality Ontologies. Newell’s [21] notion of minimal ontological commitment must be considered for this task: Only representations required to minimally describe a domain ought to be in an ontology. A micro-theory on the other hand is a formal model of knowledge required to solve a problem in a domain or describe a subset of the domain in detail [15]. It is separate from, but constructed using, general ontologies. An ontology then represents descriptions, a micro-theory, prescriptions. Therefore, ISO 9000 compliance can be automatically evaluated by using formal enterprise models based on TOVE Core and Quality Ontologies, and an ISO 9000 Micro-Theory. In the next section, the development methodology for this micro-theory is presented.

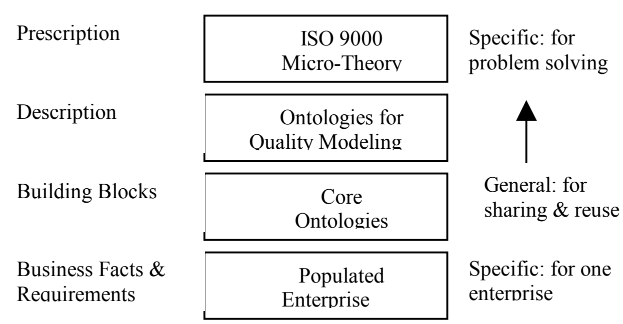


Figure 1. Model Architecture for TOVE Ontology-Based Enterprise Modeling

3. Methodology

In TOVE modeling, model builders develop ontologies and users key in business facts and requirements—i.e. they populate a model of their specific enterprise—by instantiating Core Ontologies’ terms. If an organization ‘ABC’ has an activity called ‘lathing1,’ which has a sub-activity called ‘lathing11,’ then the populated enterprise model comprises of instances like organization(ABC), activity(lathing1), activity(lathing11), and has_subactivity(lathing1,lathing11), corresponding to terms from Core Ontologies, organization(O), activity(A), and has_subactivity(A,Ao), respectively. In the Quality Ontologies, such terms are used to formally define *measure* as a *primitive measure* activity or an activity for which all its sub activities are primitive measure activities. This definition is formally expressed as follows in First-Order Logic, the representational language for TOVE Modeling.

$$\forall A \forall s\ [\ holds(\text{primitive\_measure}(A),s) \lor \forall Ao\ (\ holds(\text{has\_subactivity}(A,Ao),s) \supset holds(\text{measure}(Ao),s)\ ) \supset holds(\mathbf{measure(A)},s)\ ].$$

Primitive measure is similarly defined using Core Ontologies' terms.

Ontology expressions comprise of terms from more general ontologies or those already defined in that ontology; so too for micro-theory expressions. For example, the definition for iso_9001_4.10.1_compliant is expressed in terms of activity(A) and measure(A).

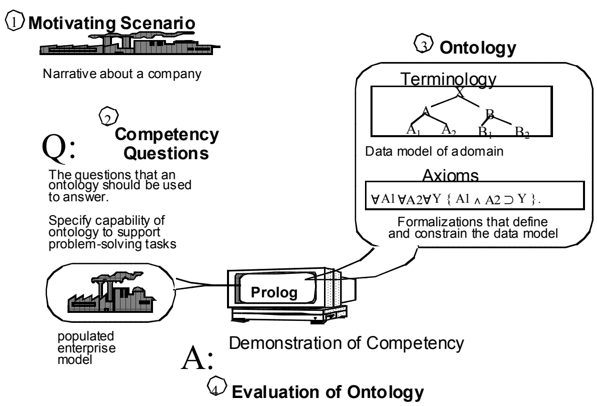


Figure 2. TOVE Ontological Engineering Methodology

A Motivating Scenario is a detailed narrative about a specific enterprise, where emphasis is placed on problems faced or tasks to perform. An IS application implements ontology based enterprise models for the problems or tasks. When the scenario is analyzed, fundamental questions about the domain independent of reference to a specific enterprise are abstracted. Given these questions, the application's requirements can be specified as Competency Questions, basic queries that it must answer, initially expressed in English. These questions focus analysis of the modeled domain through literature review, interviews, case studies, etc. The analysis' output is the Terminology comprised of the domain's key entities, relationships, and attributes. For instance, measure as an activity is identified as a key entity for the domain of quality management, and represented as measure(A) without a formal definition.

Then competency questions are expressed formally using the terminology. For TOVE, they become First-Order Logic axioms to prove. If ontology or micro-theory terminology are sufficiently defined and constrained formally as Axioms, then execution of this proof is possible. When a competency question axiom and ontology or micro-theory axioms are applied to facts represented as instances, an answer to the question is inferred if there is one. If the ontology, micro-theory, or set of instances is inaccurately stated, wrong answers are inferred; if incompletely stated, invalid null answers are inferred. Inability to infer may mean that the ontology or micro-theory is inconsistent: One ontology or micro-theory axiom contradicts another or the competency question itself. In Demonstration of Competency, the proof is executed using Prolog, which provides computer-encoded representational language and environment for automatic deduction. The implementation then:

- Demonstrates usefulness toward addressing the motivating scenario
- Verifies competency or capability of the ontology or micro-theory
- Tests accuracy, completeness, and consistency of the ontology or micro-theory

4. Ontologies

The following are excerpts of ontologies used to define ISO 9000 Micro-Theory representations. Only English definitions are presented here; Kim [20] expresses them formally.

4.1. TOVE Core Ontologies

Core Ontologies are founded upon a first-order language for representing dynamically changing worlds, called the situation calculus [24] in which each perturbation to the modeled world changes the model from one situation s to another. If a term's truth value varies in different situations, it is a fluent *f* that is said to hold (is true) in a given situation (*holds*(*f*,s)). Unless otherwise stated, all ontology terms are fluents.

An entity(X) is a tangible conceptual object class of an enterprise. The following are entities:

- resource(R) – a class of inputs and outputs of an enterprise, e.g. arm assembly.
- tru(Rt) (Traceable resource unit) – specific homogeneous collections of resources, e.g. batch 11 of arm assemblies.
- activity(A) – refers to a class or specific set of actions that transforms inputs to outputs, e.g. assemble arm assembly batch 11.
- organization_agent(O) – refers to a class of, or specifically, those that control the activity or a group of them, e.g. Fred the assembler.

An activity may actually consume a tru (consume_res_tru(A,Rt)) to produce trus of another resource (produce_res_tru(A,Rt)), and may use trus of yet another resource (use_res_tru(A,Rt)). If a tru is used, units that comprise it are not consumed so they are available for use after an activity's execution. An activity releases a tru after use. If a tru or resource is consumed/used/released/produced, the following relationship is defined: curp_res_tru(A,Rt). An activity may have sub-activities Ao (has_subactivity(A,Ao)), but a primitive activity (primitive_activity(A)) does not.

Organization agents have role(Ro)'s that they fill. These roles are filled based upon information(I), which may be a goal(G), policy(Y), or constraint(C). Information may be communicated (communication_link_of(L,I)) or authorized (authority_link_of(L,I)) using link L from source to sink roles (has_communication_source(L,Ro), has_communication_sink(L,Ro), has_authority_source(L,Ro), has_authority_sink(L,Ro)). role-info-link(Ro,L,I) represents the ternary relationship between a role, links associated with it, and the information content of that link. An authority link can initiate change of important state information about an entity (can_authorize(L,X)).

General object-oriented constructs represented are: has_subclass(X,Xo), has_instance(X,Xi), has_attribute(A,Atr), and has_attribute_value(A,Atr,V).

4.2. TOVE Ontologies for Quality Modeling: Measurement

Before quality can be evaluated, controlled, and managed, it must first be measured. Hence, measurement concepts are formalized in an ontology. In it, inspect_and_test(A) is a measure activity with special properties. Some activities are stated as control_nonconformity(A) activities. After measure, an attribute At of a tru Rt is deemed to be a conformance or nonconformance point X at time Tp (conformance_pt(X,Rt,At,Tp), nonconformance_pt(X,Rt,At,Tp)).

4.3. TOVE Ontologies for Quality Modeling: Traceability

When measurement points to a problem, traceability is the primitive analysis capability required to solve it. Hence, traceability concepts are formalized in an ontology. In it, if a tru Rt is an input into the enterprise, it is an input resource unit (input_ru(Rt)); if an output, it is an output resource unit (output_ru(Rt)). A tru that is produced then consumed within the enterprise is an intermediary resource unit (intermediate_ru(Rt)). A path L from an output resource unit to an input resource unit and intermediate paths to and from intermediate resource units can be found using a tru trace (tru_trace(Rt,Rt_1,L)).

4.4. TOVE Ontologies for Quality Modeling: Quality Management System

In order to consistently ensure that quality problems are properly measured, traced, and analyzed, there must be a quality management system (QMS) in place. Hence, QMS concepts are formalized in an ontology. In it, a quality_procedure(Y) is a quality related policy of an activity, and documented by a quality_plan(D). quality_evidence(E) is a constraint expressing whether the quality policy was followed—i.e. objective evidence. It is documented by a quality_record(D). Employee M, an individual organizational, has roless assigned to him/her (employee_has_role(M,Ro)).

4.5. TOVE Ontologies for Quality Modeling: Activity-Process Mapping Ontology

Whereas TOVE Ontologies use activity-oriented terminology—activities consume, produce, or use trus—some quality literature including ISO 9000 use process-oriented terminology—processes have inputs, outputs, and controls. Mappings between these different terminologies are formally represented in this ontology. In it, activities performed in an enterprise O are its processes A (descendent-process-organization(A,O)). Tru Rt consumed by an activity is an input into a process (process-input-tru(A,Rt)); tru produced, an output (process-output-tru(A,Rt)); and tru used, a control (process-control-tru(A,Rt)). Trus, resources, and information, all denoted by X, can be an input (process-input(A,X)), output (process-output(A,X)), or control (process-control(A,X)). An employee M may also be associated with a process (process-employee(A,M)).

4.6. Application to the Micro-Theory: Agent Constraints

An agent constraint is a special fluent, which places a constraint upon an organization agent that must be satisfied in order for that agent to achieve some goal.

holds(agent_constraint(O,c($\underline{X}$)),s)≡Φ(O,$\underline{X}$,s).

s a given situation

O an organization agent which seeks to achieve a goal in situation s

$\underline{X}$ entities and information that must be represented in order to represent the constraints on O; $\underline{X}$ is a vector with none, one, or more entities or information

c($\underline{X}$) predicate name for the agent constraint

Φ(O,$\underline{X}$,s) a first-order logic expression for the constraint named as c($\underline{X}$)

In the micro-theory, ISO 9000 compliance is represented as a goal achieved if a set of quality-related agent constraints upon an enterprise is satisfied. This goal is represented as ∃O∃s holds(agent_constraint(O,iso_9001_compliance),s), and is defined in terms of compliance to its 20 requirements like this:

∀O∀s [*holds*(**agent_constraint(O,iso_9001_compliant**),s) ≡
holds(agent_constraint(O,iso_9001_4.1_compliant),s) ∧
holds(agent_constraint(O,iso_9001_4.2_compliant),s) ∧...
holds(agent_constraint(O,iso_9001_4.20_compliant),s)].

5. ISO 9000 Micro-Theory

5.1. Motivating Scenario

> BHP Steel is an international manufacturer of quality steel products. Its Flat Products Division (FPD) produces a wide range of finished and semi-finished flat steel products from two integrated steelworks… If 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 is suspected to be an inadequate inspection system. One way to check this is to compare BHP Steel's quality inspection system with established guidelines for conducting inspection, such as the ISO 9001 requirement on inspection and testing. This check is part of BHP Steel's initiative to achieve ISO 9001 compliance.… [20]

5.2. Micro-Theory: Representing Inspection and Testing Requirements

Inspection and testing is at the core of an enterprise's capability to provide quality products. It is the last means by which nonconformities are prevented from being delivered to customers, so its importance is mentioned in the motivating scenario.

5.2.1. Competency Questions. Compliance to ISO 9001 requirement 4.10 Inspection and testing requires complying to its five sub-requirements. Thus, competency questions are informally stated as Does the company comply to: ISO 9001 requirement 4.10.1 General; 4.10.2 Receiving Inspection and testing; 4.10.3 In-process inspection and testing; 4.10.4 Final inspection and testing; and 4.10.5 Inspection and test records? These questions are represented axiomatically as ∃O∃s *holds(*agent_constraint**(**O,iso_9001_4.10.x_compliant),s)**.**

5.2.2. Terminology. ISO 9000 requirements stated in the data model are represented as agent_constraint(O, inspection_and_testing_controlled), agent_constraint(O,inspection_and_testing_recorded), agent_constraint(O,iso_9001_4.10.1_compliant), etc.

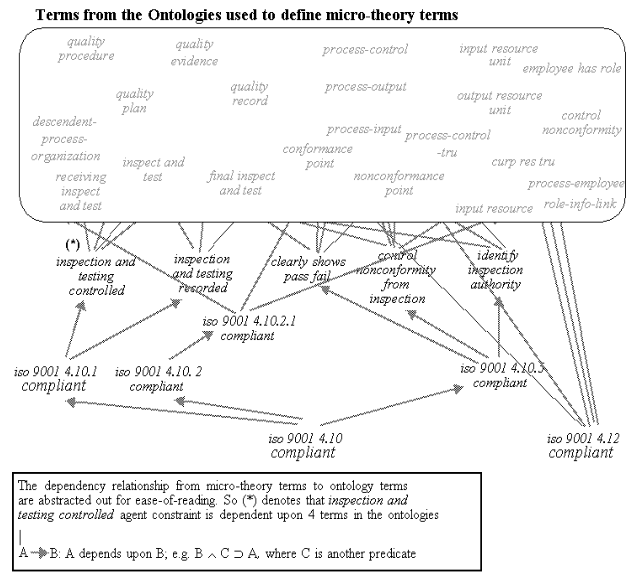


Figure 3. ISO 9000 Micro-Theory Inspection and Testing Data Model

5.2.3. Axioms. ISO 9001 requirement 4.10.1 states ([3], pp. 128]):

(i) The supplier shall establish and maintain documented procedures for inspection and testing activities in order to verify that the specified requirements for the product are met.

(ii) The required inspection and testing, and the records to be established, shall be detailed in the quality plan or documented procedures.

The micro-theory interprets (i) as: An enterprise O controls its inspection and testing in accordance with ISO 9001, if for every inspection and test process A, A is controlled by some quality procedure Ra and some documentation for the procedure, the quality plan Rb.

∀O∀A∀s [
holds(**agent_constraint(O,inspection_and_testing_controlled**),s) ≡
holds(descendent-process-organization(A,O),s) ∧
holds(inspect_and_test(A),s) ⊃
∃Ra∃Rb (*holds*(process-control(A,Ra),s) ∧
holds(process-control(A,Rb),s) ∧
holds(quality_procedure(Ra),s) ∧
holds(quality_plan(Rb),s))].

(ii) is interpreted as: An enterprise O records its inspection and testing in accordance with ISO 9001, if for every inspection and test process A, A outputs some quality evidence Rb and some quality record Ra, the documentation of this evidence.

∀O∀A∀s [
holds(**agent_constraint(O,inspection_and_testing_recorded**),s) ≡
holds(descendent-process-organization(A,O),s) ∧
holds(inspect_and_test(A),s) ⊃
RaRb (holds(process-output(A,Ra),s) ∧
holds(process-output(A,Rb),s) ∧

holds(quality_record(Ra),s) ∧
holds(quality_evidence(Rb),s))].

If (i) and (ii) are satisfied then the enterprise is compliant to 4.10.1:

∀O∀A∀s [
holds(**agent_constraint(O,iso_9001_4.10.1_compliant)**,s) ≡
holds(agent_constraint(O,inspection_and_testing_controlled),s) ∧
holds(agent_constraint(O,inspection_and_testing_recorded),s)].

Another requirement formally represented is 4.10.5, which states ([ISO 94, pp. 129]):

(iii) The supplier shall establish and maintain records which provide evidence that the product has been inspected and/or tested.[1]

(iv) These records shall show clearly whether the product has passed or failed the inspections and/or tests according to defined acceptance criteria.

(v) Where the product fails to pass any inspection and/or test, the procedures for control of nonconforming product shall apply.

(vi) Records shall identify the inspection authority responsible for the release of product.

(iv) is interpreted as: An enterprise O clearly shows pass or fail of inspection and testing in accordance with ISO 9001, if inspection and testing is recorded in accordance with ISO 9001, and for every tru Rt output from any inspection and test process A, Rt is marked as a conformance point or nonconformance point X for an attribute At, at time Tp.

∀O∀s [
holds(**agent_constraint(O,clearly_shows_pass_fail)**,s) ≡
holds(agent_constraint(O,inspection_and_testing_recorded),s) ∧
∀A∀Rt (*holds*(descendent-process-organization(A,O),s) ∧
holds(process-output-tru(A,Rt),s) ∧
holds(inspect_and_test(A),s) ⊃
∃X∃At∃Tp (
holds(conformance_pt(X,Rt,At,Tp),s) ∨
holds(nonconformance_pt(X,Rt,At,Tp),s)))].

(v) is interpreted as: An enterprise O controls nonconformities of inspection in accordance with ISO 9001, if for every tru Rt output from an inspect and test process A for which there is a nonconformance point X for attribute At at time Tp, A is an input to a control nonconforming process Ao.

∀O∀s [
holds(**agent_constraint(O,control_nonconformity_from_inspection)**,s) ≡
∀A∀Rt (*holds*(descendent-process-organization(A,O),s) ∧
holds(process-output-tru(A,Rt),s) ∧
holds(inspect_and_test(A),s) ∧
∃X∃At∃Tp *holds*(nonconformance_pt(X,At,Rt,Tp),s) ⊃
∃Ao∃s' (*holds*(process-input(Ao,Rt),s') ∧
holds(control_nonconformity(Ao),s')))].

(vi) is interpreted as: An enterprise O identifies its inspection authority in accordance with ISO 9001, if all for all trus Rt that are output from an inspection and test process A, there is an employee M with a role Ro, which has an authorization L to inspect and test Rt.

∀O∀s [
holds(**agent_constraint(O,identify_inspection_authority)**,s) ≡
∀A∀Rt [*holds*(descendent-process-organization(A,O),s) ∧
holds(process-output-tru(A,Rt),s) ∧
holds(inspect_and_test(A),s) ⊃
∃M∃Ro∃L∃(*holds*(process-employee(A,M),s) ∧
holds(employee_has_role(M,Ro),s) ∧
holds(role-info-link(Ro,L,I),s) ∧
holds(can_authorize(L,Rt),s))].

In the micro-theory, an enterprise O complies to requirement 4.10.5 if it satisfies conditions (iv) to (vi).

∀O∀s [
holds(**agent_constraint(O,iso_9001_4.10.5_compliant)**,s) ≡
holds(agent_constraint(O,clearly_shows_pass_fail),s) ∧
holds(agent_constraint(O,control_nonconformity_from_inspection),s) ∧ *holds*(agent_constraint(O,identify_inspection_authority),s)].

Though not presented here for brevity, assumptions and definitions for requirements 4.10.2 to .4 are represented in the micro-theory.

5.3. Micro-Theory: Representing Traceability Requirements

Compliance to the ISO 9000 requirement related to product identification and traceability gives confidence to an organization's customers that the organization has an adequate system to identify and locate products in various stages of production throughout the enterprise; and trace back to the cause of the nonconformity.

5.3.1. Competency Questions.

- Informally expressed: Does the enterprise comply to ISO 9001 requirement 4.8 Product identification and traceability? Formally (axiomatically) expressed: ∃O iso_9001_4.8_compliant(O).
- Informally expressed: Does the enterprise comply to the ISO 9001 requirement on product identification and classification within the enterprise? Formally (axiomatically) expressed: ∃O∃s *holds*(agent_constraint(O, product_identification_satisfied),s).
- Informally expressed: Does to the enterprise comply to ISO 9001 requirement on having product traceability capability within the enterprise? Formally (axiomatically) expressed: ∃O∃s *holds*(agent_constraint(O,traceability_satisfied),s).

5.3.2. Terminology.

[1] This requirement is subsumed by the requirement upon recording of all inspection and testing, so it is not formally represented

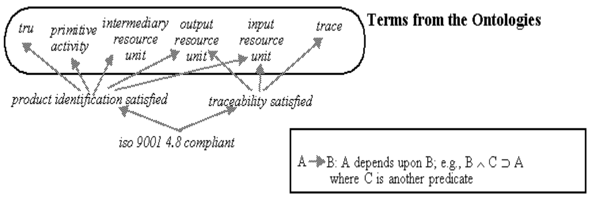


Figure 4. ISO 9000 Micro-Theory Traceability Data Model

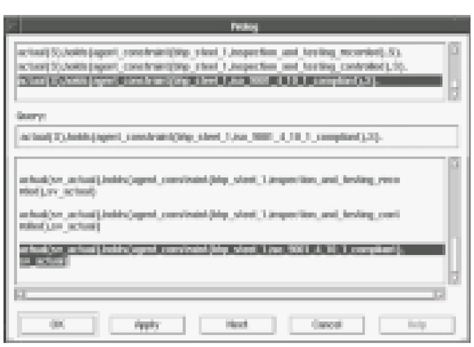
5.3.3. Axioms. In the micro-theory, an enterprise *satisfactorily identifies its products* in accordance with ISO 9001 (agent_constraint(O,product_identification)), if all trus of *primitive activities* are identified as *input*, *output*, or *intermediate resource units*. Then an enterprise complies to requirement 4.8 (agent_constraint(O,iso_9001_4.8_compliant)) if it complies to the previous requirement and the following one. Any tru Rt must be traceable back to an input resource unit Rta for a process Aa, via a trace path L_1; it must also be traceable forward to an output resource unit Rtb for a process Ab, via a trace path L_2. Violation of this requirement means that there is a tru that should, but cannot, be traced.

$\forall O \forall A \forall Rt \forall s \exists Aa \exists Ab \exists Rta \exists Rtb \exists s_o \exists s'$ [
holds(**agent_constraint(O,traceability_satisfied)**,s) ≡
holds(descendent-process-organization(A,O),s) ∧
holds(curp_res_tru(A,Rt),s) ∧ *holds*(tru(Rt),s) ∧
holds(descendent-process-organization(Aa,O),s') ∧
holds(produce_res_tru(Aa,Rta),s') ∧
holds(output_ru(Rta),s') ∧
holds(descendent-process-organization(Ab,O),s_o) ∧
holds(consume_res_tru(Ab,Rtb),s_o) ∧
holds(input_ru(Rtb),s_o) ⊃
(Rt≠Rta ⊃ $\exists L_1$ *holds*(tru_trace(Rta,Rt,L_1),s')) ∧
(Rt≠Rtb ⊃ $\exists L_2$ *holds*(tru_trace(Rt,Rtb,L_2), s_o))].

5.4. Micro-Theory: Representing Quality Management Systems Requirements

For brevity, these representations are not shown.

6. Demonstration of Competency



This is a Prolog query screen of the query evaluating BHP Steel's compliance to ISO 9001 requirement 4.10.1. Answering this query took 175 deductions.

Figure 5. Prolog Query to Answer ISO 9000 Micro-Theory Competency Question

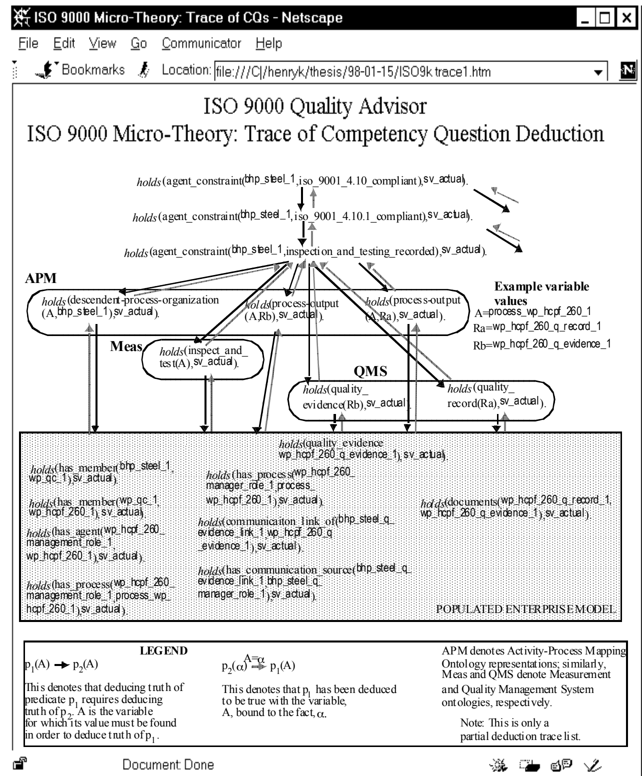


Figure 6. Graphical Display of Competency Question Deduction

7. Concluding Remarks and Future Work

The micro-theory formally represents those ISO 9000 requirements that are expressible using the Ontologies for Quality Modeling including requirements related to:

- Inspection and testing, formally represented primarily using Measurement Ontology
- Product identification and traceability, formally represented primarily using Traceability Ontology
- Management of the quality system, formally represented primarily using the Quality Management System Ontology.

The micro-theory is developed by posing competency questions, analyzing the ISO 9000 domain, and developing terminology and axioms. Then the competency of the micro-theory is demonstrated by automatically evaluating ISO 9000 compliance of a specific enterprise to a subset of requirements.

The design, analysis, and prototypical implementation of the ISO 9000 Micro-Theory demonstrates the following:

- Enterprise models based on ontologies and micro-theories can be used to perform sophisticated analysis tasks such as automated ISO 9000 compliance evaluation.
- Micro-theories formally representing other quality-related prescriptions can be reasonably developed using the Ontologies for Quality Modeling because of ontology sharability and re-usability.
- The ontology architecture and development methodology serves as a template for others' ontological engineering efforts.

This last point is especially pertinent. In the past, ontology development efforts with practical application possibilities were largely in universities (e.g Stanford's Knowledge Systems Laboratory or Toronto's TOVE), experimental (e.g. Cyc [15]), or bound in scope to classification of terms with formal semantics sparsely defined (e.g. VerticalNet ontologies [22]) when used in industry. However, XML is increasingly being adopted as a language for representing web content as a collection of values to pre-defined data definitions, i.e. as populated data models. The promise of the "Semantic Web" is to formally represent meanings of web content, i.e. as axioms that enable inference about populated models. XML enables web sharing of terminology, and the "semantic web", web sharing of semantics; they enable ontology use over the web.

Ontology use however has limitations. One is the amount of infrastructure that is required to use it. Ontology development is a form of object-oriented development, and as such, ease of development depends on the amount of existing libraries that can be used. However, developing that infrastructure can be expensive and time consuming. In fact, for one-off applications, ontology use cannot be recommended. Fortunately, the Semantic Web promises to provide a public, technical infrastructure. Also practically, some meanings of terms and conventions must be assumed to be shared or informally communicated between parties sharing an ontology. It is impossible to try to formally represent all that must be commonly understood. The difficulty lies in delineating what should be formally represented and what should be informally communicated and standardized. Finally, it still takes much training in knowledge representation and systems engineering (e.g. making that delineating decision) to develop ontologies adequately.

Nevertheless, ontology development for e-business can be widespread if Semantic Web technologies become widely adopted. TOVE ontology-based enterprise modeling is motivated by the need to solve business problems (the motivating scenario). The theme for future work is to build on lessons learned and insights from a substantial enterprise modeling exercise such as the development of the ISO 9000 Micro-Theory to offer guidance for real-life ontology development for e-business. One such application is the development of micro-theories that represent prescriptive best practices and design principles for e-business systems design. For instance, an effective website can be designed by applying rules about typical users to the sites, their needs, and how long they browse automatically to a populated model of one website design [23], to evaluate and score that design.

8. References

[1] Fox, Mark S. and Grüninger, Michael (1998). "Enterprise Modeling", *AI Magazine*, AAAI Press, Fall 1998, pp. 109-121.

[2] Campbell, A. E. and Shapiro, S. C. (1995). "Ontological Mediation: An Overview", *Proceedings of the IJCAI Workshop on Basic Ontological Issues in Knowledge Sharing*, Menlo Park CA: AAAI Press

[3] International Organization for Standards (1994). *ISO 9000 Standards for Quality Management*, Geneva: ISO Central Secretariat, 1994.

[4] Quality Digest (2000). "2000 ISO 9000 Software Buyer Guide", *Quality Digest*, July.

[5] Omni Data Sciences (1998). "The Strategic Analyst", *Omni Data Sciences*, 1345 Monroe NW, Grand Rapids, MI 49505.

[6] SAP AG (2001). "mySAP™ Product Lifecycle Management Key Capabilities—Quality Management", [Online], *Available:* *http://www.sap.com/solutions/plm/quality.htm*, May.

[7] Baan Company N.V. (2001). "iBaan Product Data Management—Product Quality and Innovation", [Online], Available: *http://www.baan.com/solutions/imake/productdata/index.html*, May.

[8] Hammer, Michael (1990). "Reengineering Work: Don't Automate, Obliterate", *Harvard Business Review*, July-August, pp. 104-112.

[9] Liles, D.H., Johnson, M.E. & Meade, L. (1996). "The Enterprise Engineering Discipline" *5th Industrial Engineering Research Conference*, Minneapolis, MN, May.

[10] Welzel, Dieter (1993). "A Rule-Based Process Representation Technique for Software Process Evaluation", *Information and Software Technology*.

[11] Grosof, Benjamin and Morgenstern, Leora (1992). "Applications of Logicist Knowledge Representation to Enterprise Modelling", *Proceedings of the AAAI-92 Workshop on Enterprise Integration*, San Jose, CA, USA, July 13.

[12] Fensel, D., Horrocks, I., Van Harmelen F., Decker, S., Erdmann, M. and Klein, M. (2000). "OIL in a nutshell" In: *Knowledge Acquisition, Modeling, and Management, Proceedings of the European Knowledge Acquisition Conference (EKAW-2000)*.

[13] Gruber, Thomas R. (1993). "A Translation Approach to Portable Ontology Specifications", *Knowledge Acquisition*, 5(2):199-220.

[14] Mahesh, Kavi & Nirenberg, Sergei. (1995) "A Situated Ontology for Practical NLP", in *Proceedings of the*

Workshop on Basic Ontological Issues in Knowledge Sharing, IJCAI-95, Montreal, Canada, August.

[15] Lenat, D. B., Guha, R. V. (1990). *Building Large Knowledge-based Systems*, Addison-Wesley.

[16] Tate, Austin (1995). "Representing Plans as a Set of Constraints - the Model", in the Special Issue on "Evaluation of Plans, Planners, and Planning Agents" of *ACM SIGART Bulletin*, Vol.6, No.1, January.

[17] Uschold, M., King, M., Moralee, S, and Yannis, Z. (1998). "The Enterprise Ontology", *The Knowledge Engineering Review*, Vol. 13.

[18] Grüninger, M., and Fox, M.S. (1995). "Methodology for the Design and Evaluation of Ontologies", *Workshop on Basic Ontological Issues in Knowledge Sharing*, IJCAI-95, Montreal.

[19] Tham, D.,Fox, M.S.,Gruninger, M. (1994). "A cost ontology for enterprise modelling, *Proceedings of third Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises*, Morgantown,WV, April.

[20] Kim, Henry M. (1999). "Representing and Reasoning about Quality using Enterprise Models", *Ph.D. Thesis*, Department of Mechanical and Industrial Engineering, University of Toronto, Toronto, Ontario CANADA M5S 3G9.

[21] Newell, A. (1982). "The Knowledge Level", *Artificial Intelligence*, 18(1), pp. 87-127.

[22] Hicks, Matt (2000). "Ontologists Give Ebiz Speech Therapy", *ZDNet eWeek*, October 16.

[23] Kim, Henry M. (2001). "Evaluating Website Effectiveness: Towards Formal Representation and Analysis of Knowledge about Design of Business Processes and Communications Structures for E-Business", *International Conference on Information Systems (ICIS)*, New Orleans LA, November, Submitted.

[24] McCarthy, J., and Hayes, P. J. (1969). "Some Philosophical Problems from the Standpoint of AI", in *Machine Intelligence 4,* Meltzer B. and Michie D. (eds.), Edinburgh, UK: Edinburgh University Press, pp. 463-501.