Ontologies and Databases – Knowledge Engineering for Materials Science Informatics

Joseph Glick

ABSTRACT

The discipline of informatics emerged from the need to translate biomedical research into evidence-based healthcare protocols and policy. Materials science informatics is rooted in an analogous need to "translate" physical sciences research and discoveries into materials-based solutions to address a broad range of issues and challenges for business, government and the environment.

Ontologies and databases are key elements of translational architectures and therefore are fundamental tools of the practice of informatics. Databases are tools for engineering data and information, while ontologies are tools for engineering knowledge and utility. Since knowledge and utility are the core objectives of informatics, correctly understanding and utilizing ontologies is critical to the development of effective materials informatics programs and tools.

Rooted in philosophy, the term ontology appears most frequently today in connection with semantic web technology, where it refers to vocabularies used by inference engines to interpret human use of language. Materials science ontologies need to capture the scientific context of the defined concepts to support modeling and prediction of multidimensional structure-property relationships in variable environments and applications.

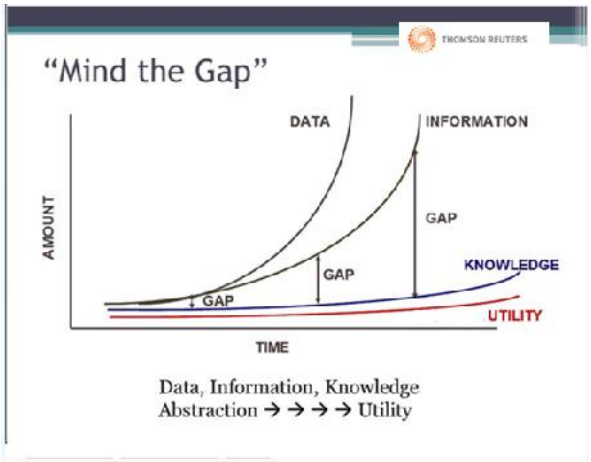
Addressing the complexity of materials science ontologies requires a significant departure from traditional database and semantic web ontology approaches, including the use of neural networks that are capable of implementing methods for modeling context, relevance, complex systems and human expertise. Pioneering efforts in this space include the Knowledge Engineering for Nanoinformatics Pilot (KENI) launched by the Nanoinformatics Society in 2010, and a collaborative Materials Genome Modeling Methodology initiative led by Iowa State University and initiated in 2011.

INTRODUCTION

Regardless of which definition we prefer, informatics is about transforming data and information into knowledge and utility. The discipline of informatics emerged in the biomedical sciences as a tool of translational medicine, the quest to transform the outcomes of basic research into generally available repositories of medical knowledge. These repositories would in turn be used by medical schools, drug manufacturers and health systems vendors to drive the development of methods and technology. It was hoped that this "translation" architecture would become a platform for evidence-based healthcare protocols and policies. Advances in genomics research, particularly the Human Genome Mapping

project, created great expectations for translational medicine. According to the National Venture Capital Association (2011) these expectations drove bio-genome related research investments to about 100 million dollars in 2000, but then the number dropped by about 80% by 2002 and stayed at approximately that level for the remainder of the decade.

The gap between the expectations and the results finds a parallel in the outcomes from bioinformatics initiatives, where even the foundational programs have made limited strides toward the driving purpose of utility, as illustrated below[1]:



The relative paucity of results cited the presentation above is relevant to materials scientists and informatics practitioners because at the time of this writing the Materials Genome Initiative promoted by the White House Office of Science and Technology Policy has been driving activities and expectations that are analogous to the bio-genomics experience. It is imperative, therefore, for materials informatics program architects and tools developers to leverage the experience of translational medicine to identify the critical issues and success factors needed to create solutions and platforms that effectively translate the outcomes of materials science research into knowledge repositories that are useful to product innovators and application engineers.

Key elements of a translational infrastructure are databases and ontologies, and although these terms frequently appear together (hence this chapter title), they represent two distinct topics with few similarities between them and many differences. Understanding the differences between databases and ontologies is imperative to the development of an effective translational architecture. The core distinction is purpose:

- Databases enable acquisition, storage, management and sharing of data and information
- Ontologies enable curation, storage, management and sharing of knowledge within contextual structures that support utility

Information is data in a useful structure. Since data and information are acquired, managed and stored in databases the first two curves in the diagram above represent the state of the database domain. The

[1] Center for Cancer Genomics and Computational Biology - Van Andel Research Institute – December 2011

exponential growth in data and information is not surprising since it is based on mature science and technology. As evidenced by the graph, a database focused approach is the simplest and shortest path to measurable outcomes, but not necessarily to valuable ones.

The legend below the graph notes that bridging the gap requires abstraction. As demonstrated by the ground-breaking predictive capabilities of the periodic table, the effectiveness of the transformations from information to knowledge is dependent on the adequacy and correctness of the abstraction methods, rather than the volume of available data or the speed of the search. Effective abstraction is a daunting challenge, especially in a computational environment, and will be discussed further in the Ontologies section that follows.

Knowledge is information in a useful context. Capturing, curating and storing knowledge is the objective of a diverse array of ontology engineering initiatives, but as the graph clearly illustrates, the limited progress made thus far in bridging the gap between computed information and knowledge is the primary obstacle to utility. A contributing factor to the gap is the dominance of semantic web ontologies among the aforementioned initiatives because they focus on standardizing vocabularies rather than the scientific context of the defined concepts.

In view of all the above, we will give primary consideration to ontologies, discussing the informatics-focused use of ontologies, the associated challenges and some sample methods that have been developed in various disciplines to address the challenges that can be leveraged by materials science. Thereafter we will review the roles and limitations of databases, with a special emphasis on the "big data" environment with is gaining the lion's share of today's information mind space. In conclusion, we will review some of the pioneering initiatives that are utilizing the approaches and methods that will be discussed in this chapter, including a materials science example.

ONTOLOGIES

Coined by Aristotle to name the search to understand the nature of things, the term "ontology" resided primarily in the domain of philosophy until the emergence of the semantic web technology. The semantic web relies heavily on structured vocabularies, commonly referred to as ontologies, to define concepts and relationships. These ontologies are usually application-focused, defining the terms and conceptual structure relationships that the application will need to properly interpret the natural language user inputs, with the fields of medicine and pharmacology leading the way in the early years in terms of number of initiatives and the allocation of funds. The rapidly growing role of search engines and social media sites and applications has expanded the range of players, with marketing initiatives emerging as the dominant divers of the new growth in semantic web ontology projects.

Standards for semantic web technologies and methods are developed by w3.org. As the W3C website explains[2], "There is no clear division between what is referred to as "vocabularies" and "ontologies". The trend is to use the word "ontology" for more complex, and possibly quite formal collection of terms,

[2] http://www.w3.org/standards/semanticweb/ontology

whereas “vocabulary” is used when such strict formalism is not necessarily used or only in a very loose sense. Vocabularies are the basic building blocks for inference techniques on the Semantic Web.”

This lack of clear distinction has not been a serious concern for the semantic web community, but for materials scientists it is a profound issue with far reaching consequences, not the least of which is the risk of failure to maintain a distinction between that which is learned from evidence (science) and that which is decided upon by consensus (negotiated or voted standards). Materials science informatics requires the use of both ontologies and vocabularies to engineer solutions to grow the body of useful knowledge related to materials and their characteristics. However, both ontologies and vocabularies must fulfill one or more of an array of precisely defined roles, and each role needs to be enabled by appropriately optimized architectures supported by technologies that are capable of implementing the required architectures.

Ontologies, Vocabularies and Materials Science Informatics

As evidenced by the White House Office of Science and Technology Policy Materials Genome Initiative, the development of economic opportunities and efforts to address environmental and health issues that could be solved or attenuated with new materials are driving interest and investment in the field. To align with these demands, the purpose of materials science informatics ontologies can be defined as a set of three concrete objectives:

1. Translate data and information into knowledge that is useful, not only to materials scientists, but also to application engineers, environmentalists, regulators and other users
2. Curate the knowledge base to maintain alignment with emerging scientific research and discovery, as well as with application development or engineering innovation across the relevant disciplines and industries
3. Present the knowledge in a flexible architecture that supports tailoring the outputs based on multiple criteria to make the utility and value correctly understandable to each user group

Information is data that is captured and stored in a useful structure, and knowledge is information that is modeled and presented in a useful context. Therefore, translating materials data and information into useful knowledge requires ***defining and modeling the scientific context of a library of concepts*** to support the exploration and prediction of multidimensional structure-property relationships in variable environments and applications. The scientific context of a concept should certainly include the definition and classification attributes of the concepts (vocabularies and taxonomies), but achieving the desired prediction capabilities requires modeling and quantifying structural and behavioral attributes, as well as the potential combinatorial interactions between and across the concepts and attributes. Furthermore, to enable model-based experimentation and discovery, the ontology must reside in an architecture that supports the coexistence of an expanding array of models, each of which behaves as an independent agent, supported by technologies that enable hitherto unknown or at least unspecified chaotic and stochastic interactions between and across the models.

The above described computational environment is most likely unfamiliar to many scientists and informatics practitioners because the tools and systems with which they may be acquainted are

architected around data and information. However, all will intuitively recognize that the conceptual description of the model-based experimentation and discovery ontology architecture is representative of the real world problems with which we are wrestling. Let's consider a drug safety example.

The FDA Center for Drug Evaluation and Research requires the calculation of embedded uncertainty in various models as part of the risk assessment process[3]. CDER will consider whether there is a biologically plausible explanation for the association of the drug and the safety signal, based on what is known from systems biology and the drug's pharmacology. Their web site states: "The more biologically plausible a risk is the greater consideration will be made to classifying a safety issue as a priority. We must assume three models to analyze uncertainty: 1) A dynamical model that predicts the consequences for specific parameters, 2) a hierarchical model that describes population variability between individuals and 3) a measurement model that describes how observations, including errors, were made. We can be uncertain about any of the three models as well as the parameters that describe those models."

In the real world the various models created independently to support the analysis are interdependent at both the model and attribute levels. This is an example of systemic complexity, which will be discussed further in the Challenges and Methods section below. However, the capability to model the complex interactions implicit in the evaluation description by the FDA is fundamental to translating real world data and information into real world knowledge.

Since in theory computer systems are supposed to model the real world, the fact that familiar computational scenarios are quite dissimilar from familiar real world scenarios may provide a clue as to why the gap between information and knowledge continues to grow. The science and business of informatics need to supplement the existing data and information focused computing environments with knowledge and utility focused computing architectures and technologies if we are to begin narrowing the widening gap.

Curating the knowledge base has its own special needs, including but not limited to:

- A registry of source databases containing research data, standardized vocabularies and other relevant content needs to be developed and maintained.
- Connections and / or update processes need to be developed between the source databases and the ontology architecture.
- Updates, analysis, review, and implementation processes need to be designed; and automated to the extent resources and technology permit.

In the first bullet we have an example of a precisely defined role for multiple vocabularies. In this use case - curating your knowledge base - the vocabularies are being developed, curated and stored outside your materials science informatics ontology architecture, and in most cases they are not materials science vocabularies. However since no science exists in a vacuum, even when a researcher's interests are very narrowly focused, the scientific context of the concepts modeled in your ontology will include and are dependent upon components of knowledge from other disciplines.

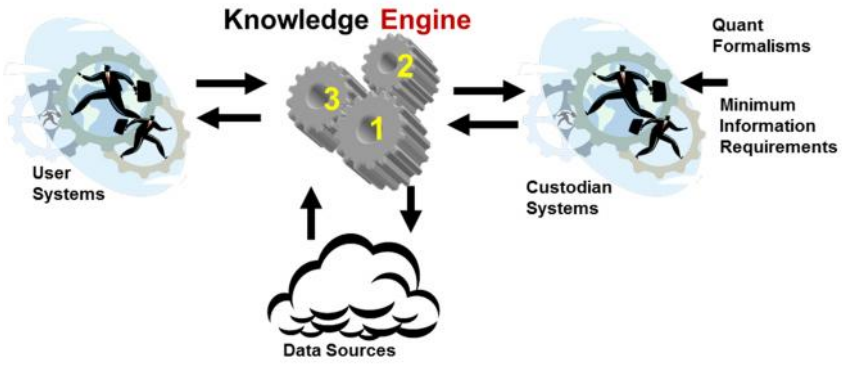
[3] http://www.fda.gov/Drugs/DrugSafety/default.htm

Let's consider the example of the development of nanomaterials for use in drug delivery systems. The physico-chemical characterization context will include information that resides in physics and chemistry databases. The pharmacological characterization context will include information that resides in pharmaceutical, biomedical, and biochemical databases. In each relevant database the information is defined and classified using discipline-specific vocabularies and taxonomies. The mappings between the potentially relevant vocabularies and the core vocabulary of the materials science ontology need to be maintained to support the processes described in the second and third bullets.

Utility requires knowledge to be presented in a generally available form that enables its potential users to find relevant content, correctly interpret the content, and utilize it effectively. While there are many potential presentation approaches, a critical element of delivering unambiguous tailored outputs is the inclusion of the vocabularies normally used by the user groups (another example of a specific role for vocabularies) and mapping them to the materials science vocabulary at the core of the ontology. The navigation use cases would allow the users to intuitively identify their disciplinary and problem domain parameters because they are navigating in their own language, and the internal engine would then perform the relevance computations required to deliver the tailored results.

It is clear that attempting to deliver on the above objectives is not without its significant challenges, and therefore easy to see why the gap between information stores and useful knowledge continues to grow.

The goal, however, is not out of reach. The first step is defining an appropriate solution architecture, as in the example below from the Knowledge Engineering for Nanoinformatics Pilot[4] which will be discussed later in this chapter:



The major components of the Knowledge Engine support the core abstractions of the content domains. The components, represented by the yellow numbers on the gears, are:

1. Scientific Ontology – contextually modeled concepts, attributes and interactions
2. Vocabularies and Mappings – core and related based on utility scope
3. Analysis, Experimentation and Presentation Rules – including user navigation use cases

[4] www.nanoinformatics.org

The legend below the gap between information and knowledge graph in the Introduction section presents abstraction as a critical requirement for the development of solutions to bridge that gap. The above architecture abstracts the content simultaneously by nature (i.e. knowledge, terminology and rules) and by purpose (i.e. to compute context, to translate meaning and to enable use cases). The result not only maintains the distinction between the science which is learned from evidence and the vocabularies which are decided by consensus, but also keeps both content groups segregated from the implementation methodologies which will vary according to the specific project requirements.

The evolution of this abstraction approach is further discussed in the Recent Initiatives section below, which also includes details of a sample materials science implementation of the architecture. The section that follows will discuss the more significant of the computational challenges, and will present examples of emerging methods that have been leveraged to address them. Since most of the described methods have been developed outside of materials science informatics, the examples have been selected based on what seems to illustrate the approach in the most generic way. The aim is to facilitate understanding of the underlying principles that can then be applied in materials science informatics tools and applications.

Challenges and Methods

The scale and persistence of the gap between information stores and knowledge bases provides evidence that the obstacles that need to be overcome are not trivial. The challenges presented here to bridging the gap are not exhaustive by any means, since there are social, academic, economic and other issues that are clearly outside the scope of this work.

In the context of informatics knowledge acquisition, curating and sharing are the primary objectives. Therefore, the primary perspective we will use to analyze the challenges and solution methods associated with bridging the gap is that of knowledge engineering. This discipline, which is rooted in the integration of computer science innovation with cognitive science discoveries, leverages proven engineering methodologies to integrate knowledge into computer systems in order to solve complex problems normally requiring a high level of human expertise. It is an emerging discipline with most references to it occurring within academic contexts, but there have been practical applications, as shown in some of the examples and in the Recent Initiatives section at the conclusion of this chapter.

From the knowledge engineering perspective, there are three primary challenges:

1. Complexity
2. Relevance computation
3. Capturing human expertise as a computable asset

We will discuss each one individually, and consider methods that have had at least some measure of success in dealing with these issues.

Complexity

In the context of building computational solutions the term “complexity” is frequently, although not always precisely, used in three common senses:

1. Mathematical complexity which from a computer science perspective refers to the rules that define the way the scaling of data drives the scaling of computational resource requirements. Mathematical complexity is directly related to the volume of the data, and this is a dominant topic today across disciplines and domains due to the publicity and expectations surrounding Big Data processing platforms. We will review the current state of this technology, including what it can and cannot do for materials science informatics, in the Big Data subsection of the Databases section below.
2. Design complexity from a solution architecture perspective, which when used precisely refers to the challenges associated with implementing and scaling highly flexible and dynamic processes. When used imprecisely, especially as a criticism of a computational approach, the term complexity usually refers to complications, the errors and inefficiencies created when systems and/or components are integrated and patched together ad nauseum without due attention to problem analysis or solution architecture and reengineering.
3. Systemic complexity from a real-world modeling perspective, which refers to the chaotic and stochastic interactions within, between, and across physical, biological and social systems. Systemic complexity has received some recent notoriety in technology circles due to the significance of its role in the global financial crisis.

All three issues are important to the goal of transforming information into knowledge, and all these issues are addressed by the approaches and methods being utilized and under development within the knowledge engineering discipline. Modeling systemic complexity beyond the most limited scope, however, includes the challenges associated with mathematical and design complexity, and since systemic complexity is the most relevant to materials science ontology engineering we will review the complexity issue from the systemic perspective.

Scientists working to model environmental and biological systems are very cognizant of the challenges, but the commonly used computational tools and technologies are very limited when facing scaling systemic complexity, resulting in the creation of a diverse array of knowledge silos, each limited by the scalability of the supporting technology. While recognizable utility can be achieved in the implementation of educational use cases such as Wolfram Alpha[5] and the NanoHub[6], the lack of dynamic interaction between the silos is a significant obstacle to tackling broader and more critical research and discovery objectives. Even when they do their own coding, scientists understandably depend on technology vendors to provide the tools that they need, and the awareness of the limitations of generally used tools to address scaling systemic complexity is just beginning to emerge within the computing vendor community.

To assist vendors in grasping this error, and the significance of the systemic complexity management issue, DARPA launched its "Real-World Reasoning" project in 2005 (internally called "Get Real"). The report in COMPUTERWORLD begins[7]:

[5] www.wolframalpha.com
[6] www.nanohub.org
[7] Gary H. Anthes DECEMBER 05, 2005 (COMPUTERWORLD)

> December 5, 2005 (COMPUTERWORLD) – It is surely one of the more mind-blowing PowerPoint slides ever created. It's a graph, and one of the smallest numbers, near the bottom of the vertical axis, is 10^{17}, the number of seconds from now until the sun burns up. Then comes 10^{47}, the number of atoms on Earth. After that, the numbers get really big, topping the scale at $10^{301,020}$.
>
> This graph, from the Defense Advanced Research Projects Agency, shows the exponential growth in possible outcomes for a range of activities, from a simple car engine diagnosis with 100 variables to war gaming with 1 million variables (that's what the $10^{301,020}$ represents).
>
> The point DARPA is trying to make in explaining the Real-World Reasoning Project is that computers will never be able to exhaustively examine the possible outcomes of complex activities, any more than a roomful of monkeys with typewriters would ever be able to re-create the works of Shakespeare.

In the real world, human judgment and expertise rule when problem domains are fraught with significant complexity or uncertainty, and whether the organizations involved are government, business, military, or altruistic the highest authority and compensation are given to those perceived as effective decision makers. After receiving a DARPA "Real World Reasoning" grant in 2008 to begin research in this area, IBM acknowledged, "Today's computers are powerful number crunchers but don't do a good job of dealing with ambiguities or integrating information from multiple sources into a holistic picture of an event."[8]

The combinatorial complexity issue targeted by the DARPA initiative is driving IBM and five academic partners, who jointly won the funding, to develop a neural chip[9], abandoning the traditional silicon chip design for a more biomimetic architecture. The goal is not just faster processing, but also the scaling of current capacity for combinatorial computation. Researchers at MIT have taken a different approach, seeking to mimic the brain's plasticity by modeling the activity of a single synapse using about 400 transistors[10]. While both efforts are significant from a research perspective, for technology managers charged with enabling advanced analytics capabilities within their organizations, or for scientists and engineers charged with delivering the next generation of informatics platforms, the research initiatives described above serve only to emphasize the challenges ahead. Not the least of the challenges is thinking about what kind of data and information structures will be required to support biomimetic processing architectures and technologies.

Systemic Complexity Modeling Methods

Since the consensus from the above described initiatives is that a biomimetic approach is required to model systemic complexity, it is reasonable to look to cognitive science and biologically inspired design methods for insight into how real world reasoning would represent complex systems. In this subsection

[8] InformationWeek November 20, 2008 IBM Eyes Computers That Mimic The Brain

[9] 8/24/2011 http://www.computerworld.com/s/article/print/9219288/IBM_brings_brain_power_to_exp...

[10] http://www.mit.edu/newsoffice/2011/brain-chip-1115.html

we focus on four principal methods for modeling systemic complexity – cognitive architectures, capturing and automating elements of human expertise, relevance computation and using "Models as Agents", an agent-based modeling approach in which individual models and their components are behaving as independent agents capable of chaotic and stochastic behaviors.

Cognitive Architectures

The development and use of cognitive architectures is about leveraging the insights that have been enabled by brain scanning technology about the information architecture in the human brain. From a cognitive science perspective, the requirements for effective representation for complex domains are well understood, and they are[11]:

1. Integrate levels of abstraction – this may seem like an obvious requirement, but it isn't
2. Combine globally homogeneous with locally heterogeneous representation of concepts
3. Integrate alternative perspectives of the domain
4. Support malleable manipulation of expressions
5. Have compact procedures
6. Have uniform procedures

To integrate levels of abstraction the modeling team needs to first identify all the levels of abstraction within the scope of the domain and make sure that they are precisely distinguished and represented. For example, it may be intuitive to identify systems and structures within the domain as subdomains, but behavioral categories across domains need to be linked to the properties within the domains with which they interact.

An example of combining globally homogeneous with locally heterogeneous representation of concepts is object oriented inheritance, which is a useful tool in modeling systems. The aim is to define a simple concept structure into which all the members of your highest level of abstraction can fit comfortably. If this goal continues to be elusive after serious effort, then there may be higher levels of abstraction in your domain than the ones with which you are working.

Integrating alternative perspectives of the domain means creating a context representation architecture which allows concepts to be interpreted differently and model elements to behave differently based on contextual drivers and parameters. The design and construction of context engines is not trivial, and a critical success factor is a sufficiently robust assumptions layer that is aggregated dynamically from properties and their values across elements of the domain, as well as including external interaction scenarios.

The capability to support malleable manipulation of expressions is required so that the context architecture can be flexible, since methods and values in expressions will need to vary according to the driving scenario. Some of the specific methods for achieving this type of flexibility are:

- Include variables in the appropriate expressions that are dependent on your data and updates

[11] *P.C.-H. Cheng /Cognitive Science 26 (2002) 685–736*

- Include calculated variables in the appropriate expressions that are outputs of other expressions in your rules engine or procedures
- Create expression libraries that are available for inclusion into your procedures based on the computed context
- Enable users to make contextual inputs and decisions at run time

Having compact procedures supports the creation of a procedure library that is managed by the context engine and therefore allows the procedures to be assembled and ordered dynamically as required by the driving process outputs and user selections at run time.

Having uniform procedures supports the use of engineering techniques to optimize the sharing and reuse of procedures and expressions. Using this approach enables more precise and streamlined abstraction of the algorithmic architecture, in addition to supporting improved performance and simplified maintenance.

A core principle of cognitive architecture is that human knowledge is abstracted into three fundamental categories – semantic, episodic and procedural. To advance usability, which requires aligning computation with user mental models, each concept class in our scientific ontology should have a *cognitive* state (Semantic, Episodic or Procedural). For example, the concept "inhibition" could apply to:

1. The effect of a particle on its target, or the environment on the particle – semantic
2. Experimental data, or patent prior art – episodic
3. A biological pathway, or a lab process – procedural

Each unique combination of keyword and cognitive state is a distinct class. The cognitive state of the concept class is defined at build time, and would then become an input to the concept relevance computation discussed below.

Human Expertise

Capturing human expertise requires a significant departure from current tools and technical approaches that are labeled "knowledge management," due to the significant differences between the problem-solving approaches of experts and those of educated novices.

Studies of syntactic, semantic, schematic, and strategic differences in problem analysis and solution approaches between recent graduates with advanced degrees and recognized experts in physics, computer science and medicine revealed the following common, distinguishing characteristics of experts[12]:

1. Rapidly and effortlessly recognize issues and anomalies
2. Work with mental models that connect observations and input
3. Manipulate large clusters of information based on context
4. Analyze and plan abstractly and consider many alternatives

[12] Mayer, R., 1992, *Thinking, Problem Solving, Cognition,* Freeman

As an example, human experts capable of playing blind chess (without looking at physical pieces on a board) do not necessarily have a superior memory (like a computer), but have a much more extensive repository of scenarios and associated rules that fill in what "must be" based on relevant context anchors[13].

The cognitive behavior of professional novices in each of the four above areas was the inverse. Therefore the experts' cognitive behavior requires an abstract modeling environment to define their problem solving context, as well as the ability to specify discrete, coexisting scenarios associated with the context. This captured expertise can then be delivered to novices to help solve problems.

Modeling the cognitive behavior of experts requires software that does not limit experts to the application designers' view of the world. They need the power to define qualitative, flexible contexts, as well as frameworks and rules for how information is interpreted as the contexts are evolved. Traditional software and data architecture cannot meet this challenge because the highly constrained data structures and deterministic use case driven algorithms do not have the capability to implement the cognitive architectures described in the above section, or the relevance computation requirements that are discussed below. The key technical reasons for these limitations will be considered in the Databases section later in the chapter.

Since a human user needs to act on the output, the utility of the knowledge is dependent on the alignment of the output with the user's mental models, especially when the objectives are decision quality and learning. Cognitive research[14] has demonstrated that:

- the definition of the learning objective is not based solely on the accuracy of knowledge, but also on the subjectively and contextually determined utility of knowledge being acquired
- humans entertain multiple hypotheses and learn not only by modifying a single existing hypothesis but also by combining a set of hypotheses

The conclusions above are intuitive and are as applicable to decisions as to learning, but the issue of utility is frequently ignored and is a major cause of the too common disconnect between user needs and the functionality of delivered systems.

Relevance Computation

Relevance analysis and computation addresses combinatorial complexity because it is the foundation of real-world reasoning. For decades, cognitive researchers have understood that flexible mental models created by contextual and subjective relevance processes enable the superior analytical and decision capabilities of experts in comparison with those of educated novices[15]. Additionally, recent research has shown that when asked to sort descriptions of real-world phenomena, novice students of the physical

[13] Reinhold Behringer. "Augmented Reality." In Allen Kent and James G. Williams, eds., *Encyclopedia of Computer Science and Technology*, Vol. 45, No. 30, pp. 45-57. Marcel Dekker, Inc., 2001

[14] Toshihiko Matsuka et al, Neurocomputing - August 2008

[15] Mayer, R., 1992, *Thinking, Problem Solving, Cognition*, Freeman

sciences sorted primarily by the domain, whereas experts sorted primarily by causal category[16], emphasizing that effective relevance analysis integrates procedural and episodic knowledge with the semantic approach to which the domain categorization is limited. As a result we can see from another perspective why a strictly semantic ontology is inadequate for capturing scientific knowledge, since the causal relationships are so critical for the understanding and predictability of behaviors and interactions.

The conclusions above are intuitive and are as applicable to decisions as to learning, but as mentioned earlier, the issue of utility is frequently ignored and is a major cause of the too common disconnect between user needs and the functionality of delivered systems. Consequently, addressing the challenge of utility includes enabling the following technological capabilities:

- a contextual architecture that supports the definition and simultaneous interaction of multiple hypotheses and abstraction methods
- a relevance computation engine that can link the properties and attributes of the hypotheses to data across domains and levels of abstraction
- analysis and maintenance processes across the layers of context and hypothesis that are driven by a combination of information updates, computations, and dynamic input from curators or users

A contextual architecture clearly needs to be a multidimensional structure which should be a core element of the design of the scientific ontology. The dimensions could include, but are not limited to:

- Domain
- Problem
- Scope
- Scale
- Properties
- Property Values
- Methods
- Behaviors
- Formalisms
- Models

The construction of a contextual architecture may seem to be a threat to the host system's capacity to handle the algorithmic complexity because the effort will capture significant complexity in the modeled environment. Staged processing will offset the threat by computing in stages and at each stage filtering for relevance, and only computing those paths found to be relevant, thus minimizing the computing resources required. The more precise and granular the context architecture, the more efficiency will be gained by staged relevance computation, resulting in a synergistic relationship between the structural and algorithmic relevance methods.

Models as Agents

To align with real world complexity modeling requirements and methods described above, knowledge engineering solutions need to be able to receive expert input in a way that captures their mental models

[16] B. M. Rottman et al, 2011, Causal Systems Categories: Differences in Novice and Expert Categorizations of Causal Phenomena

of relevance within and across target semantic domains, together with their contextual and subjective associations with procedural components and episodic (historic) data. The captured mental models acting as agents in deterministic and stochastic interactions form the basis for real-world reasoning networks that can assist with the key challenges described.

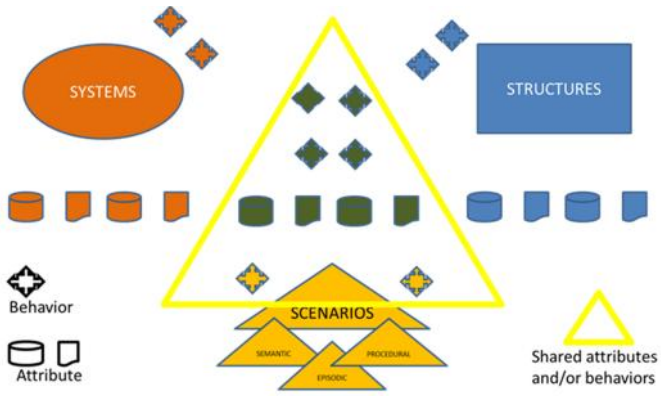
In the Models as Agents approach, data points are modeled as a neural network of interacting systems, structures and scenarios, as well as their attributes and behaviors. The scenarios drive relevance computation, and:

- are composed of semantic, procedural and/or episodic elements
- can be situated across systems and structures (scenarios share behaviors with the systems and structures)
- may be emergent behaviors of the systems and structures (scenarios share attributes with the systems and structures)

Scenario attributes include relational, hierarchical, unstructured and random data. Scenario elements compute relevance, and are abstracted as:

- Ontologies
- Taxonomies
- Categories (Chomsky)
- Agents (including external models)
- Rules

The holistic environment can be illustrated graphically as follows:



Attributes of systems, structures and scenarios are defined as categories mapped across relational, hierarchical, unstructured, and random data sources. Behaviors of systems, structures and scenarios are defined as expressions that include static and / or dynamic variables and operators.

This real-world reasoning approach enables the construction of models that integrate highly diverse elements and information sources to enable exploration and discovery to a scope that traditional information architecture cannot accommodate.

DATABASES

It is assumed that the readers have used and likely created a variety of databases during the course of their education, career and perhaps other pursuits, and it is not the purpose of this text to teach the fundamentals of computer science. Therefore we will focus our review of database technology in this section on the roles and limitations of databases from an analytics and informatics perspective. In recent years, the emphasis and expectations associated with "big data" in literature, marketing and professional activities have dominated the computing mind share in academic, business and government circles. Therefore we will give the majority of our attention in this section to understanding the current and emerging technologies, architectures, and analytical approaches in this field, as well as where it fits in the goal of informatics to bridge the gap between information and knowledge.

Roles

Information is data in a useful structure, which means that the structure allows computer algorithms to access the data in a way that enables the program to accomplish the task for which it was designed. The basic functionality of a database is data storage with read, write and edit capabilities. Commercial database products are engineered to provide additional functionality around that structure based on what the designers understand to be useful to the both the human and systemic actors that will be connected to the database. That additional functionality includes, but is not limited to, processes and tools for multiple user access capability, access control, security, configuration, integration, customization, administration and procedures used by end users. The vast majority of the Total Cost of Ownership (TCO) of a database product – purchase price, infrastructure, administration and maintenance, etc. – is driven by the additional functionality and is designed around the requirements of enterprise applications. Informatics requirements are not identical to those of enterprise computing, and we will address this in the Big Data subsection below.

The roles of a database can therefore be summarized as:

1. Store data
2. Maintain data
3. Enable users and programs to access the data

Limitations

The structure of today's commercially available database products is relational. From a scientific perspective, however, that designation can be misleading, especially if people refer to databases and ontologies in the same context. Even when the use of the term ontology is limited to the structured vocabularies of the semantic web, the relationships that are defined (e.g. subsumption) are meaningful semantic relationships between the connected concepts. A relational database structure allows developers to define links between data tables by defining key fields and "relating" them to fields in other tables, but these links provide no insight as to the real world relationships between the information in the linked tables.

What the links do create are potential pathways that algorithms or queries can use to navigate between tables, and yet this capability is also a key limitation of relational databases from an analytics and

informatics perspective. These pathways are very analogous to roads and highways that are designed for automobile traffic. They enable standard vehicles (not off road) to get from point A to point B, but the path is not always the most direct geographically, and you cannot drive to a location to which there is no road. In the same way, the fundamental limitations of relational databases when it comes to transforming information into knowledge are:

1. The methods for abstracting, processing, and querying the data are limited to what is permitted by the database schema
2. Relationships between data elements that are not on a defined path are invisible to algorithms

The impact of these limitations when working with a single database is that you can only use the data in ways that have been foreseen by the database designers, and you can only query for what you know is there. However when multiple data sources are integrated into a single information infrastructure, the result is the creation of information silos that present serious obstacles to knowledge capture and discovery.

It is precisely these limitations of databases that have driven the development of search technologies and engines. However, the effectiveness of search is highly dependent on the ability of the searcher to describe precisely what they are seeking. The more advanced search engines attempt to perform relevance analysis and developers continue working on improving the semantic inference engines.

From an informatics perspective, search engines are an end user use case for a scientific ontology which combines semantic, procedural and episodic context computation, and can be used to output tailored search parameters for targeted problems and user groups.

Big Data

According to Wikipedia[17], "In information technology, big data is a collection of data sets so large and complex that it becomes difficult to process using on-hand database management tools. The challenges include capture, curation, storage, search, sharing, analysis, and visualization." At the time of this writing, a Google search on "big data" yields over two billion results, and according to IBM[18] "Every day, we create 2.5 quintillion bytes of data — so much that 90% of the data in the world today has been created in the last two years alone."

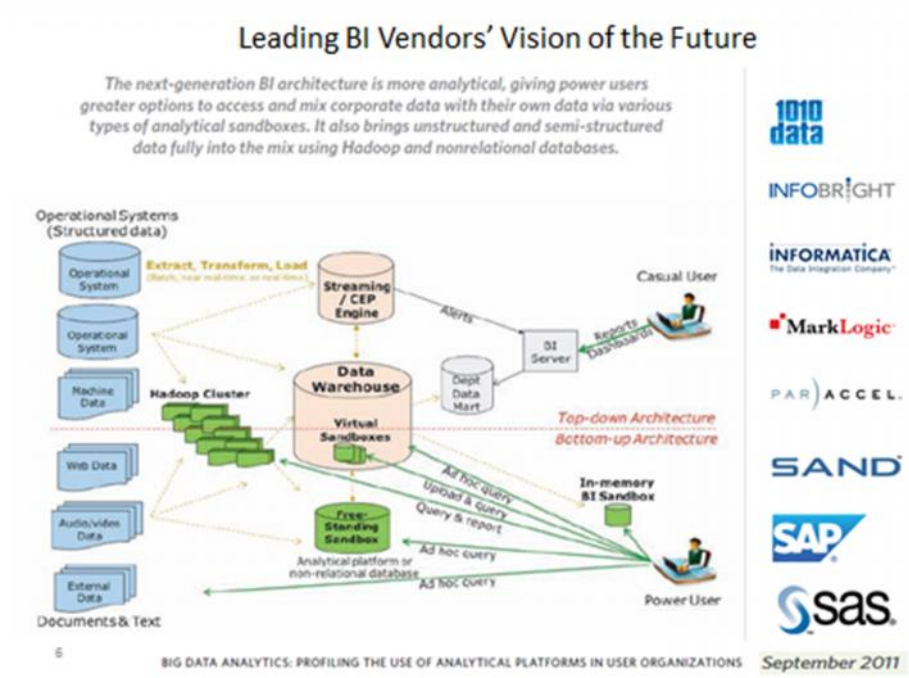
It is therefore not surprising that science, business and government are struggling with the daunting challenges of transforming rapidly scaling stores of "big data" into evidence-based, actionable intelligence. The limited achievements and unrealized expectations recorded thus far by leading enterprises[19] underscore the obstacles to interdisciplinary knowledge integration, which is critical to discovery and value.

As discussed in the introduction to this chapter, the efforts by informatics professionals to translate the exponential growth in data volume into information and utility have met with very limited success. The

[17] http://en.wikipedia.org/wiki/Big_data
[18] http://www-01.ibm.com/software/data/bigdata/
[19] http://www.zdnet.com/fords-big-data-chief-sees-massive-possibilities-but-the-tools-need-work-7000000322

fact that global enterprises with much larger budgets than informatics scientists find themselves in the same predicament underscores the gravity of the challenges discussed in this chapter. We also reviewed above the limitations of traditional database technology, and how these limitations not only constrain knowledge capture and discovery, but create information silos within integrated environments, and this is a particularly critical issue with big data environments because the data from operational systems that populate the massive parallel processing platform may have undergone various transformations across integrated systems.

Business Intelligence (BI) vendors are very conscious of these limitations, since the larger ones are also database product developers. Currently, they are seeking to leverage the "sandbox" data architecture approach utilized effectively by the developers of geographical positioning software, to enable complex, cross-platform query applications. The key strategy involves combining the sandbox architecture with in-memory analytics tools to break through the existing information barriers built over decades by traditional system architecture implementations, as seen below:



In the diagram above from a whitepaper funded by the listed vendors, the green components represent the future, but rather than depicting any new technology the architecture envisions new products and services being developed as an additional layer of infrastructure to enable queries across information silos. Nevertheless, there remain major limitations to the above described approach to acquiring knowledge by combing data from diverse sources and running queries:

1. The presented approach does not address the combinatorial complexity issue which was discussed in the systemic complexity section above.
2. Queries have a very limited capacity for knowledge acquisition and discovery.

Addressing Impacts of Complexity

When it comes to attenuating the impacts of complexity, the sandbox architecture is important to the relevance computing approaches presented earlier in this chapter because the data structure is optimized for holistic access to data from multiple sources by algorithms and query tools. However, the context engines discussed earlier which are needed to implement the cognitive methods for relevance computation require something more - to access data stored in neural sandboxes. In a neural structure, the data from multiple sources can not only be stored and accessed holistically, but it can also be abstracted across multiple dimensions, including the creation of categories across data types and dynamic categories that are dependent on values and formula outputs.

Creating the above described data environment enables a powerful method for addressing computational complexity – ***staged processing integrated with relevance computation***. As the scope of the data scales, the combinatorial complexity is attenuated by the use of neural networks to minimize data point and content redundancy and to enable the reuse of links, relationships, formalisms and other components. The analytical engine can compute in stages and at each stage filter for relevance, and then only needs to compute those paths found to be relevant, thus minimizing the computing resources required. This approach can yield significant results for both the end user analytic capability architecture (more details are set forth in the Knowledge Acquisition and Discovery heading below) and the massive parallel processing architecture.

In a big data scenario, this approach can be used as follows:

1. Define a statistically valid sample of the source data sets and output to the neural sandbox.
2. Perform the staged processing integrated with relevance computation on the sample data and identify the relevant paths.
3. Create an output file to be imported, or systematically pass the relevant paths specification to the massive parallel processing platform (e.g. a Hadoop cluster) to process the available data more efficiently or to increase the viable scope of the data processed.

The value of this approach scales with the complexity of the data, because parallel processing requires breaking up the data into sections, creating in effect information silos within the processing architecture. If the data is very homogenous, then that may not be a problem. However, in a more typical scenario as illustrated in the diagram above, the complexity of the data will require multiple iterations punctuated by analysis activities because each processing node is looking at a small subset of the data.

To help deal with multiple data sources and types, some vendors have built, and others are building, a mapping engine. The above described approach could be used to improve the precision and streamline the maintenance of the mappings.

Knowledge Acquisition and Discovery

Just as the large database vendors understand the limitations of their products, they also understand that queries are very limited tools when it comes to knowledge acquisition and discovery. Jill Dyché, the

Vice President of Thought Leadership at SAS, wrote[20]: "The common thread running through many of big data's most promising explorations is discovery. Traditional database inquiry requires some level of hypothesis, but mining big data reveals relationships and patterns that we didn't even know to look for…These patterns are too specific and seemingly arbitrary to specify, and the analyst would be playing a perpetual guessing-game trying to figure out all the possible patterns in the database. Instead, special knowledge discovery software tools find the patterns and tell the analyst what--and where--they are." She then sites as an example the fact that researchers at Stanford University were mining data on breast cancer cells expecting to see trends in cell proliferation rates. But, to their surprise, they discovered that surrounding non-cancerous cells are also contributing to cancer cell growth. The researchers who made this discovery didn't know to look at the non-cancerous cells. But through low-hypothesis exploration, they found it.

It is clear that traditional database inquiry is not enough, but the very example cited above shows that simply identifying patterns is not capturing knowledge or discovery. Finding the patterns is an observation which is only the first step of the scientific method. That observation leads to theories (e.g. correlation versus causality) which need to be tested and iteratively refined by experimentation and analysis, and the farther along that our computational environment advances the process the more real knowledge acquisition and discovery is achieved.

On the other hand, using staged processing integrated with relevance computation in a desktop in-memory environment allows analysts to do much more than observe patterns (which is clearly important). The process works as follows:

1. The analysts are presented navigation options that reflect the context architecture, allowing the experts to specify what is relevant to their problem domain.
2. The selections are translated into relevant analysis paths and passed to the Hadoop cluster.
3. The relevant data is assembled and output to the neural sandbox.
4. The experts dynamically define scenarios and navigate all the relevant data, testing the value and validity of the patterns, and exploring causal relationships, not just content similarities.

Criticality of Relevance Computation in Big Data

When the volume and complexity of the data puts exhaustive computation out of reach, the necessity of computing relevance is evident. However, relevance computation is not just a fallback position to cope with scaling data, but is usually the critical path to an effective solution. For example, the expression "real-time decision support" is appearing with increasing frequency in discussions of "big data" issues and objectives, but in the real world it is often an oxymoron. Real-time information is only actionable when it reports events for which all the following conditions exist:

1. the event is known and has been analyzed
2. a policy exists (i.e. a decision has already been made) for responding
3. a process exists to respond
4. resources have been allocated to respond

[20] http://blogs.hbr.org/cs/2012/11/eureka_doesnt_just_happen.html

A current study found that delivering operational information to Mobile BI users reduced average decision time from 190 hours to 66 hours[21]. While the improvement is significant, the results show that after receiving the information, users still needed to make a significant investment of time and effort to achieve utility – i.e. make a decision, and we have no data on how frequently the users decided that the output did not really address the problem at hand.

Relevance analysis is a core element of problem analysis, which is all about asking the right questions, so if a real-time solution is being considered, then the first question is: are we seeking to enable informed decisions or automate implementation of decisions already made? If large and complex data stores are involved, the solutions for decision support and event management are usually mission-critical and expensive propositions, and as a result relevance analysis is a critical success factor.

The impact of the relevance criticality is:

- the ***utility*** of information delivered for both discovery (research, event tracking, etc.) and analysis (decision support, planning, etc.) is directly proportional to the *granularity and precision of the context architecture*
- the ***total cost of ownership (TCO)*** is inversely proportional to the *human expertise that is embedded* in the relevance analysis processes

Consequently, to the extent that context can be precisely architected and relevance correctly computed, it increases the value and utility of the outputs and reduces the TCO. This is possible because problem analysis becomes an embedded driving force in operational and maintenance processes, not just a high-level exercise at the beginning of a project. System architects are well aware of the cost-of-errors heuristic: errors not detected at the requirements stage cost ten times as much to repair at the design and construction stage, and ten times more at the testing stage. Defining requirements to solve the wrong problems can add orders of magnitude to this painful reality, but relevance analysis is the critical path to ensure that the solution assembles the correct data and methods to transform information into knowledge, and knowledge into utility.

Addressing the challenges of complexity, expertise and dynamic scalability requires a new paradigm, because as Einstein well put it, “You cannot solve problems using the thinking that caused them.”

Recent Initiatives

The focus that the White House Office of Science and Technology Policy (OSTP) has placed on materials science as a potential key player in the efforts toward economic recovery has stimulated the creation of many new Materials Genome initiatives, as well as the rebranding of existing ones. Most of these, however, have limited themselves to the Data and Information domains. Two exceptions are the Knowledge Engineering for Nanoinformatics Pilot (KENI) launched by the Nanoinformatics Society and a collaborative Materials Genome Modeling Methodology initiative led by Iowa State University.

[21] March 2012, Mobile BI 2012: Accelerating Business on the Move, Aberdeen Group

Knowledge Engineering for Nanoinformatics

Funded by the NSF, the Nanoinformatics Society held its first conference in November 2010 with the aim of bringing together a multidisciplinary group of experts and stakeholders to address the informatics need of Nanotechnology. One of the workshop groups was tasked with scoping a pilot to address the utility of data and information related to nanotechnology across a broad variety of potential users. The group decided that this project *would not be an exercise in consensus arbitration over definitions and mappings*. Instead, the goal was to make validated knowledge and disciplinary expertise understandable and computable so that the cross-disciplinary value can be made available and usable to an array of stakeholders. Achieving this goal requires the definition of a cognitive framework and information architecture that is free from disciplinary and technological bias. The KENI pilot addresses this requirement by means of Computable Context Representation (CCR).

CCR is an innovative methodology which the KENI Pilot is using to model the complex and dynamic relationships between the inputs (user types, disciplinary domains, analysis purpose, type & scope, etc.) and the outputs (ontology engineering[22] architectures, relevant data and sources, parameters, quantitative methods, etc.).

The primary innovative value driver is the combination of:

- Defining a network of qualitative and quantitative concepts, the relevance of which is computed based on user inputs
- Formalizing, and where possible, quantifying the interactions between concepts
- Leveraging cognitive architectures and principles to guide network design and relevance computation

The initial focus was on toxicity prediction by applying the approach to linking chemistry, structure and biology datasets. These efforts were still underway at the time of this writing.

While all knowledge engineering efforts seek to incorporate elements of cognitive science, a key aspect of CCR innovation is the driving role of a cognitive architecture that will be supported by appropriate information architectures. The chart below summarizes the selected methods, with the cognitive focus highlighted in the blue box.

Mitigation Strategies	Chaos	Opaqueness	Silos	Complexity	Uncertainty
Neural Modeling	X		X	X	
Rule Inference		X		X	
Relevance Inference	X		X	X	X
Interaction Simulation	X	X	X	X	X
Cognitive Architectures	X			X	X
Conceptual Rationalization	X		X	X	
Scenario Automation				X	X

[22] Rajan, K., ONTOLOGY ENGINEERING: COMPUTATIONAL INFORMATICS FOR AN ICME INFRASTRUCTURE (2011)

Rather than being an infrastructure-centric solution, the KENI is a portable and extensible architecture which can be deployed within existing infrastructures, thus accelerating the path to utility and value realization. There are many initiatives seeking to bridge the gap between massive and rapidly scaling data stores and the potential value to be derived, and the pilot team created this summary of highly visible initiatives (IBM Watson, LarKC and Wolfram Alpha) and the approaches taken in comparison to Expertool knowledge engineering approach used by the KENI Pilot:

	IBM Watson	Expertool	LarKC	Wolfram Alpha
Computes	Answer	Relevance	Selected process outputs	Answer
Approach	- Statistical match - Computing power (85,000 watts)	- Contextual parsing - Concept quantification - Relationship discovery	-Massive, distributed and incomplete reasoning lab	- Linguistic parsing - Curating computable knowledge

By 2011 the KENI pilot gained momentum, absorbed the activities of some other pilots, and was divided into three subprojects, including the materials science focused Quantitative Structure Activity Relationship (QSAR) subproject. The outgrowth of the latter is the Iowa State University initiative that is described in the following subsection.

Materials Genome Modeling Methodology at Iowa State University

The Combinatorial Sciences and Materials Informatics Collaboratory (CoSMIC) is an international collaborative research program focused on data driven discovery in materials science using a genomics discovery paradigm for materials design. Its central research theme is to develop new computational and experimental ways of accelerated mechanistic based discovery and design of materials using informatics methods. The program is directed by Professor Krishna Rajan of Iowa State University and involves a network of laboratories in over ten countries.

On November 14, 2011 the White House Office of Science and Technology Policy announced on its blog under the heading “Mapping the Materials Genome” as follows[23]:

> Iowa State University, Los Alamos National Laboratory, and Ames Laboratory, in partnership with a network of universities and industrial partners, will be initiating a series of workshops starting in 2012 called “Mapping the Materials Genome”. These meetings are focused on identifying the critical research challenges and establishing the experimental and computational techniques by which the “Materials Genome” can in fact be realized. Activities will also include short courses and educational materials to establish a network for training the scientific workforce with the skills in “Materials Genomics”.

Among the computational techniques referred to above are the approaches based on knowledge engineering that are described in this chapter, and they are being implemented and tested within the Materials and Omics Modeling Platform research program jointly sponsored by CoSMIC and The Expertool Paradigm, LLC. This program is led by Professor Rajan and the author of this chapter, with

[23] http://www.whitehouse.gov/blog/2011/11/14/support-grows-president-obama-s-materials-genome-initiative

materials science and informatics expertise supplied by Iowa State and knowledge engineering expertise and software tools provided by Expertool. The first output of this new program is a proof of concept model to demonstrate the applicability of the knowledge engineering methods to the materials science domain. This proof of concept model is described in the following subsection, first from a materials science perspective and then from a knowledge engineering perspective. In the subsections that follow, we will summarize how the methods described in this chapter were implemented to deal with the key challenges of addressing complexity, computing relevance as well as capturing and automating human expertise.

Materials Science Proof of Concept Model

The materials science domain selected for this proof of concept by the Iowa State team was apatite crystal chemistry, and input data for the model was selected based on review of relevant literature. For the selection of the compounds and the crystal symmetry (space group) of the compounds the literature references are:

Kendrick and Slater (2008) Mat Res. Bull. 43, 2509-2513.
Kendrick and Slater (2008) Mat Res. Bull. 43, 3627-3632.
Kendrick and Slater (2008) Sol State Ionic 179, 981-984.
Leon-Reina et al. (2003) Chem Mater 15, 2099-2109
Leon-Reina, L. et al. (2005) Chem. Mater. 17, 596-601
Orera et al. (2011) Fuel Cells. 11 10-16.
Pramana et al. (2008) J. Sol. State. Chem. 181, 1717-1722.
Sansom et al. (2005) Sol. State Ionics 176, 1765-1769
Rabe et al. (1992) Phys. Rev. B 45 7650-7676.
Shannon, R. D. (1976) Acta Cryst. A 32 751-767.

The input data for each compound was the following:

- At each of three sites (Lanthanum, Germanium and Oxygen), a quantitative measurement of each of four properties (*Zunger's Pseudopotential Core Radii Sum, Martynov-Batsanov Electronegativity, Valence Electron Number* and *Shannon's Ionic Radius*) for a total of eleven measurements per compound (one property, Shannon's Ionic Radius, did not apply at the Oxygen site).
- A qualitative classification of crystalline structure (P-1, P63/m, or no apatite).

The data was organized into a table with each compound as row; each measurement was a column, text columns for the structure class and literature reference, as shown below:

	Compound	Zunger's Ps	Martynov-B	Valence Ele	Zunger's Ps	Martynov-B	Valence Ele	Zunger's Ps	Martynov-B	Valence Ele	Shannon's I	Shannon's I	Crystal Sy	Literature R
1	La9.5Ge4.5Al1.5O25.5	2.926	1.283	2.85	1.589	1.903	3.75	0.439	3.136	5.667	1.155	0.39	No apatite	Leon-Reina, L
2	La9.5Ge5Al1O25.75	2.926	1.283	2.85	1.579	1.932	3.833	0.443	3.166	5.722	1.155	0.39	No apatite	Leon-Reina, L
3	La9.4Ge5.5Al0.5O25.85	2.895	1.269	2.82	1.57	1.961	3.917	0.445	3.179	5.744	1.143	0.39	No apatite	Leon-Reina, L
4	Nd9.33Ge6O26	3.723	1.12	2.799	1.56	1.99	4	0.448	3.197	5.778	1.085	0.39	P-1	Orera et al. (
5	Pr9.33Ge6O26	4.18	1.026	2.799	1.56	1.99	4	0.448	3.197	5.778	1.1	0.39	P-1	Orera et al. (
6	La8Sr2Ge6O26	3.106	1.306	2.8	1.56	1.99	4	0.448	3.197	5.778	1.235	0.39	P63/m	Pramana et a
7	Nd8Sr2Ge6O26	3.834	1.186	2.8	1.56	1.99	4	0.448	3.197	5.778	1.192	0.39	P63/m	Orera et al. (
8	Pr8Sr2Ge6O26	4.226	1.106	2.8	1.56	1.99	4	0.448	3.197	5.778	1.205	0.39	P63/m	Orera et al. (
9	La8Ba2Ge6O26	3.144	1.296	2.8	1.56	1.99	4	0.448	3.197	5.778	1.267	0.39	P63/m	Kendrick and
10	La9.5Ge5.5Al0.5O26	2.926	1.283	2.85	1.57	1.961	3.917	0.448	3.197	5.778	1.155	0.39	P63/m	Leon-Reina, L
11	La9.33Ge4Ti2O26	2.874	1.26	2.799	1.9	1.947	4	0.448	3.197	5.778	1.135	0.4	P63/m	Sansom et al
12	La10Ge4Ga2O26	3.08	1.35	3	1.605	1.893	3.667	0.448	3.197	5.778	1.216	0.417	P-1	Kendrick and
13	La8Ba2Ge4Ti2O26	2.804	1.188	2.6	1.9	1.947	4	0.448	3.197	5.778	1.12	0.4	P63/m	Sansom et al
14	La8.67BaGe4Ti2O26	3.011	1.278	2.801	1.9	1.947	4	0.448	3.197	5.778	1.201	0.4	P63/m	Sansom et al
15	La8Y2Ge4Ga2O26	3.052	1.362	3	1.605	1.893	3.667	0.448	3.197	5.778	1.188	0.417	P63/m	Kendrick and
16	La9.33Si2Ge2Ti2O26	2.874	1.26	2.799	1.853	1.943	4	0.448	3.197	5.778	1.135	0.357	P63/m	Sansom et al
17	La9.33Ge3Ti3O26	2.874	1.26	2.799	2.07	1.925	4	0.448	3.197	5.778	1.135	0.405	No apatite	Sansom et al
18	La9.33Ge3Ti3O26	2.874	1.26	2.799	2.07	1.925	4	0.448	3.197	5.778	1.135	0.405	No apatite	Sansom et al
19	La9.33Ge3Ti3O26	2.874	1.26	2.799	2.07	1.925	4	0.448	3.197	5.778	1.135	0.405	No apatite	Sansom et al
20	La9.6Ge5.5Al0.5O26.15	2.957	1.296	2.88	1.57	1.961	3.917	0.45	3.215	5.811	1.167	0.39	P63/m	Leon-Reina, L
21	La8.4Ba1.6Ge6O26.2	3.132	1.307	2.84	1.56	1.99	4	0.451	3.222	5.822	1.257	0.39	P63/m	Kendrick and
22	La8Y2Ge4.4Ga1.6O26.2	3.052	1.362	3	1.596	1.913	3.733	0.451	3.222	5.822	1.188	0.411	P63/m	Kendrick and
23	La10Ge4.5Ga1.5O26.25	3.08	1.35	3	1.594	1.918	3.75	0.452	3.228	5.833	1.216	0.41	P-1	Kendrick and
24	La9.67Ge5.5Al0.5O26.25	2.978	1.305	2.901	1.57	1.961	3.917	0.452	3.228	5.834	1.176	0.39	P-1	Leon-Reina, L
25	La9.52Ge6O26.28	2.932	1.285	2.856	1.56	1.99	4	0.453	3.231	5.84	1.158	0.39	P63/m	Leon-Reina e
26	La9.54Ge6O26.31	2.938	1.288	2.862	1.56	1.99	4	0.453	3.235	5.847	1.16	0.39	P63/m	Leon-Reina e
27	La8.55Y1Ge6O26.33	2.927	1.295	2.865	1.56	1.99	4	0.453	3.238	5.851	1.147	0.39	P63/m	Kendrick and
28	La6.55Y3Ge6O26.33	2.899	1.307	2.865	1.56	1.99	4	0.453	3.238	5.851	1.119	0.39	P63/m	Kendrick and
29	La9.56Ge6O26.34	2.944	1.291	2.868	1.56	1.99	4	0.454	3.239	5.853	1.162	0.39	P63/m	Leon-Reina e
30	La9.58Ge6O26.37	2.951	1.293	2.874	1.56	1.99	4	0.454	3.243	5.86	1.165	0.39	P63/m	Leon-Reina e
31	La9.75Ge5.5Al0.5O26.375	3.003	1.316	2.925	1.57	1.961	3.917	0.454	3.243	5.861	1.186	0.39	P-1	Leon-Reina, L
32	La9.6Ge6O26.4	2.957	1.296	2.88	1.56	1.99	4	0.455	3.246	5.867	1.167	0.39	P63/m	Leon-Reina e
33	La8.8Ba1.2Ge6O26.4	3.119	1.318	2.88	1.56	1.99	4	0.455	3.246	5.867	1.246	0.39	P63/m	Kendrick and
34	La8Y2Ge4.8Ga1.2O26.4	3.052	1.362	3	1.587	1.932	3.8	0.455	3.246	5.867	1.188	0.406	P63/m	Kendrick and
35	La8.63Y1Ge6O26.45	2.952	1.306	2.889	1.56	1.99	4	0.456	3.252	5.878	1.157	0.39	P63/m	Kendrick and
36	La7.63Y2Ge6O26.45	2.938	1.312	2.889	1.56	1.99	4	0.456	3.252	5.878	1.143	0.39	P63/m	Kendrick and
37	La6.63Y3Ge6O26.45	2.924	1.318	2.889	1.56	1.99	4	0.456	3.252	5.878	1.129	0.39	P63/m	Kendrick and
38	La9.8Ge5.5Al0.5O26.45	3.018	1.323	2.94	1.57	1.961	3.917	0.456	3.252	5.878	1.192	0.39	No apatite	Leon-Reina, L
39	La9.66Ge6O26.49	2.975	1.304	2.898	1.56	1.99	4	0.456	3.257	5.887	1.175	0.39	P-1	Leon-Reina e
40	La9SrGeO26.5	3.093	1.328	2.9	1.56	1.99	4	0.456	3.259	5.889	1.225	0.39	P-1	Pramana et a
41	La10Ge5Ga1O26.5	3.08	1.35	3	1.583	1.942	3.833	0.456	3.259	5.889	1.216	0.403	P-1	Kendrick and
42	La9BaGe4Ti2O26.5	3.112	1.323	2.9	1.9	1.947	4	0.456	3.259	5.889	1.241	0.4	P63/m	Sansom et al
43	La9BaSi2Ge2Ti2O26.5	3.112	1.323	2.9	1.853	1.943	4	0.456	3.259	5.889	1.241	0.357	P63/m	Sansom et al
44	La9.68Ge6O26.52	2.981	1.307	2.904	1.56	1.99	4	0.457	3.261	5.893	1.177	0.39	P-1	Leon-Reina e
45	La9.7Ge6O26.55	2.988	1.31	2.91	1.56	1.99	4	0.457	3.265	5.9	1.18	0.39	P-1	Leon-Reina e
46	La8.71Y1Ge6O26.57	2.977	1.317	2.913	1.56	1.99	4	0.458	3.267	5.904	1.167	0.39	P63/m	Kendrick and
47	La7.71Y2Ge6O26.57	2.963	1.323	2.913	1.56	1.99	4	0.458	3.267	5.904	1.153	0.39	P63/m	Kendrick and
48	La6.71Y3Ge6O26.57	2.949	1.329	2.913	1.56	1.99	4	0.458	3.267	5.904	1.138	0.39	P63/m	Kendrick and
49	La9.72Ge6O26.58	2.994	1.312	2.916	1.56	1.99	4	0.458	3.268	5.907	1.182	0.39	P-1	Leon-Reina e
50	La9.2Ba0.8Ge6O26.6	3.106	1.328	2.92	1.56	1.99	4	0.458	3.271	5.911	1.236	0.39	P63/m	Kendrick and
51	La8Y2Ge5.2Ga0.8O26.6	3.052	1.362	3	1.578	1.951	3.867	0.458	3.271	5.911	1.188	0.401	P63/m	Kendrick and
52	La9.74Ge6O26.61	3	1.315	2.922	1.56	1.99	4	0.458	3.272	5.913	1.184	0.39	P-1	Leon-Reina e
53	La9.75Ge6O26.625	3.003	1.316	2.925	1.56	1.99	4	0.459	3.274	5.917	1.186	0.39	P-1	Leon-Reina e
54	La7.55Y2Ge6O26.33	2.913	1.301	2.865	1.56	1.99	4	0.459	3.275	5.918	1.133	0.39	P63/m	Kendrick and
55	La8.79Y1Ge6O26.69	3.001	1.328	2.937	1.56	1.99	4	0.46	3.282	5.931	1.176	0.39	P63/m	Kendrick and
56	La7.79Y2Ge6O26.69	2.987	1.334	2.937	1.56	1.99	4	0.46	3.282	5.931	1.162	0.39	P63/m	Kendrick and
57	La6.79Y3Ge6O26.69	2.973	1.34	2.937	1.56	1.99	4	0.46	3.282	5.931	1.148	0.39	P63/m	Kendrick and
58	La9.4Ba0.6Ge6O26.7	3.099	1.334	2.94	1.56	1.99	4	0.46	3.283	5.933	1.231	0.39	P63/m	Kendrick and
59	La9.6Ba0.4Ge6O26.8	3.093	1.339	2.96	1.56	1.99	4	0.462	3.295	5.956	1.226	0.39	P-1	Kendrick and
60	La8Y2Ge5.6Ga0.4O26.8	3.052	1.362	3	1.569	1.971	3.933	0.462	3.295	5.956	1.188	0.395	P63/m	Kendrick and
61	La8.87Y1Ge6O26.81	3.026	1.338	2.961	1.56	1.99	4	0.462	3.297	5.958	1.186	0.39	P63/m	Kendrick and
62	La7.87Y2Ge6O26.81	3.012	1.344	2.961	1.56	1.99	4	0.462	3.297	5.958	1.172	0.39	P63/m	Kendrick and
63	La6.87Y3Ge6O26.81	2.998	1.35	2.961	1.56	1.99	4	0.462	3.297	5.958	1.158	0.39	P63/m	Kendrick and
64	La8.95Y1Ge6O26.93	3.051	1.349	2.985	1.56	1.99	4	0.464	3.311	5.984	1.196	0.39	P63/m	Kendrick and
65	La7.95Y2Ge6O26.93	3.037	1.355	2.985	1.56	1.99	4	0.464	3.311	5.984	1.182	0.39	P63/m	Kendrick and
66	La6.95Y3Ge6O26.93	3.023	1.361	2.985	1.56	1.99	4	0.464	3.311	5.984	1.168	0.39	P63/m	Kendrick and
67	La10Ge6O27	3.08	1.35	3	1.56	1.99	4	0.465	3.32	6	1.216	0.39	P-1	Pramana et a
68	La10Ge6O27	3.08	1.35	3	1.56	1.99	4	0.465	3.32	6	1.216	0.39	P-1	Pramana et a
69	La10Ge6O27	3.08	1.35	3	1.56	1.99	4	0.465	3.32	6	1.216	0.39	P-1	Pramana et a
70	Nd10Ge6O27	3.99	1.2	3	1.56	1.99	4	0.465	3.32	6	1.163	0.39	P-1	Orera et al. (
71	Pr10Ge6O27	4.48	1.1	3	1.56	1.99	4	0.465	3.32	6	1.179	0.39	P-1	Orera et al. (
72	La8Yb2Ge6O27	3.182	1.3	3	1.56	1.99	4	0.465	3.32	6	1.181	0.39	P63/m	Orera et al. (
73	La8Yb2Ge6O27	3.182	1.3	3	1.56	1.99	4	0.465	3.32	6	1.181	0.39	P63/m	Orera et al. (
74	La8Yb2Ge6O27	3.182	1.3	3	1.56	1.99	4	0.465	3.32	6	1.181	0.39	P63/m	Orera et al. (
75	La8Gd2Ge6O27	3.246	1.3	3	1.56	1.99	4	0.465	3.32	6	1.194	0.39	P-1	Orera et al. (
76	La8Sm2Ge6O27	3.292	1.32	3	1.56	1.99	4	0.465	3.32	6	1.199	0.39	P-1	Orera et al. (
77	La8Nd2Ge6O27	3.262	1.31	3	1.56	1.99	4	0.465	3.32	6	1.205	0.39	P-1	Orera et al. (
78	Nd8Y2Ge6O27	3.78	1.242	3	1.56	1.99	4	0.465	3.32	6	1.145	0.39	P-1	Orera et al. (
79	Pr8Y2Ge6O27	4.172	1.162	3	1.56	1.99	4	0.465	3.32	6	1.158	0.39	P-1	Orera et al. (
80	La6Y4Ge6O27	3.024	1.374	3	1.56	1.99	4	0.465	3.32	6	1.16	0.39	P-1	Orera et al. (
81	La9.83Ge5.5Nb0.5O27	3.028	1.327	2.949	1.66	1.993	4.083	0.465	3.32	6	1.195	0.398	P63/m	Orera et al. (
82	La9.83Ge5.6Nb0.4O26.95	3.028	1.327	2.949	1.64	1.993	4.067	0.464	3.314	5.989	1.195	0.396	P63/m	Orera et al. (
83	La9.83Ge5.7Nb0.3O27	3.028	1.327	2.949	1.62	1.992	4.05	0.463	3.308	5.978	1.195	0.395	P63/m	Orera et al. (
84	La9.83Ge5.8Nb0.2O27	3.028	1.327	2.949	1.6	1.991	4.033	0.462	3.302	5.967	1.195	0.393	P63/m	Orera et al. (
85	La9.83Ge5.9Nb0.1O27	3.028	1.327	2.949	1.58	1.991	4.017	0.462	3.295	5.956	1.195	0.392	P63/m	Orera et al. (
86	La10Ge5.5Ti0.5O27	3.08	1.35	3	1.645	1.979	4	0.465	3.32	6	1.216	0.393	P-1	Orera et al. (
87	La8Y2Ge5Ti1O27	3.052	1.362	3	1.73	1.968	4	0.465	3.32	6	1.188	0.395	P63/m	Kendrick and
88	La10Ge4.5Ti1.5O27	3.08	1.35	3	1.815	1.958	4	0.465	3.32	6	1.216	0.398	P63/m	Orera et al. (
89	La9Y1Ge6O27.005	3.066	1.356	3	1.56	1.99	4	0.465	3.32	6	1.202	0.39	P63/m	Kendrick and
90	La8Y2Ge6O27.005	3.052	1.362	3	1.56	1.99	4	0.465	3.32	6	1.188	0.39	P63/m	Kendrick and
91	La7Y3Ge6O27.005	3.038	1.368	3	1.56	1.99	4	0.465	3.32	6	1.174	0.39	P63/m	Kendrick and
92	La10Ge5.9W0.1O27.1	3.08	1.35	3	1.58	1.987	4.033	0.467	3.332	6.022	1.216	0.391	P63/m	Orera et al. (
93	La10Ge5.8W0.2O27.2	3.08	1.35	3	1.599	1.983	4.067	0.468	3.345	6.044	1.216	0.391	P63/m	Orera et al. (
94	La10Ge5.7W0.3O27.3	3.08	1.35	3	1.619	1.98	4.1	0.47	3.357	6.067	1.216	0.392	P63/m	Orera et al. (
95	La10Ge5.6W0.4O27.4	3.08	1.35	3	1.638	1.977	4.133	0.472	3.369	6.089	1.216	0.392	P63/m	Orera et al. (
96	La10Ge5.4W0.6O27.6	3.08	1.35	3	1.678	1.97	4.2	0.475	3.394	6.133	1.216	0.393	No apatite	Orera et al. (
97	La10Ge5.3W0.7O27.7	3.08	1.35	3	1.697	1.967	4.233	0.477	3.406	6.156	1.216	0.394	No apatite	Orera et al. (
98	La10Ge5.2W0.8O27.8	3.08	1.35	3	1.717	1.963	4.267	0.479	3.418	6.178	1.216	0.394	No apatite	Orera et al. (
99	La10Ge5.4Nb0.6O27.3	3.08	1.35	3	1.68	1.994	4.1	0.47	3.357	6.067	1.216	0.399	No apatite	Orera et al. (
100	La10Ge5.35Nb0.65O27.3	3.08	1.35	3	1.69	1.994	4.108	0.471	3.36	6.072	1.216	0.4	No apatite	Orera et al. (
101	La10Ge5.3Nb0.7O27.3	3.08	1.35	3	1.7	1.995	4.117	0.471	3.363	6.078	1.216	0.401	No apatite	Orera et al. (
102	La10Ge5.5W0.5O27.5	3.08	1.35	3	1.658	1.973	4.167	0.474	3.381	6.111	1.216	0.393	P63/m	Orera et al. (

The first step of modeling the data set was to specify formulas to dynamically compute the maximum and minimum values of each measurement within each crystal symmetry structural category, along with

the maximum and minimum values for the entire sample, or for those of a user selected subset of the sample. Since the calculations were set to be dynamic, the maximum and minimum values would be updated if any of the source data was refreshed with changed values, or if new rows with additional compounds were added to the table.

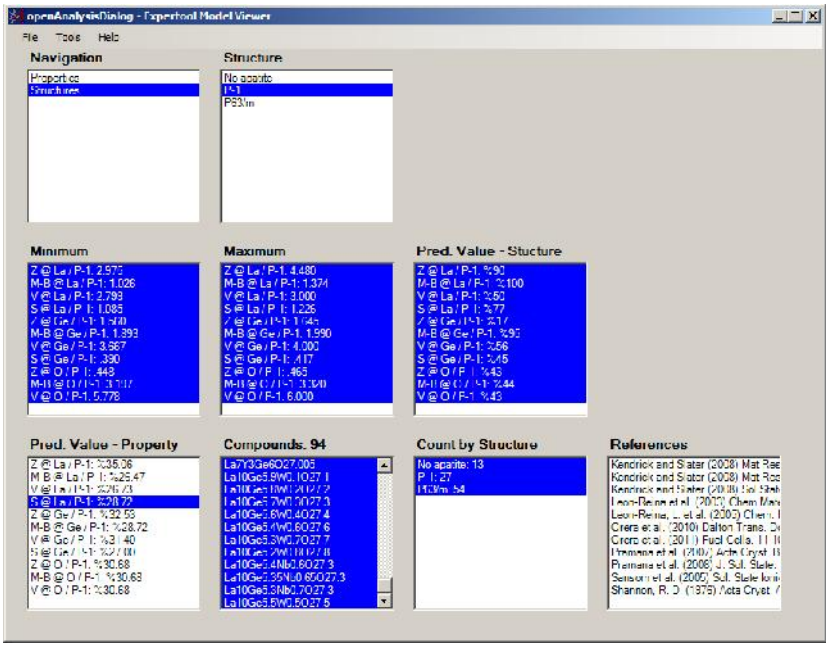
The calculation of the maximum and minimum value outputs enabled a user to browse for structure / measurement pairs where the range was small compared to that for the entire sample. This indicated how consistent a predictor the structural category was of the quantitative measurement.

The next step of constructing the model was to list each combination of structure, site and measurement and to set each combination as a results category.

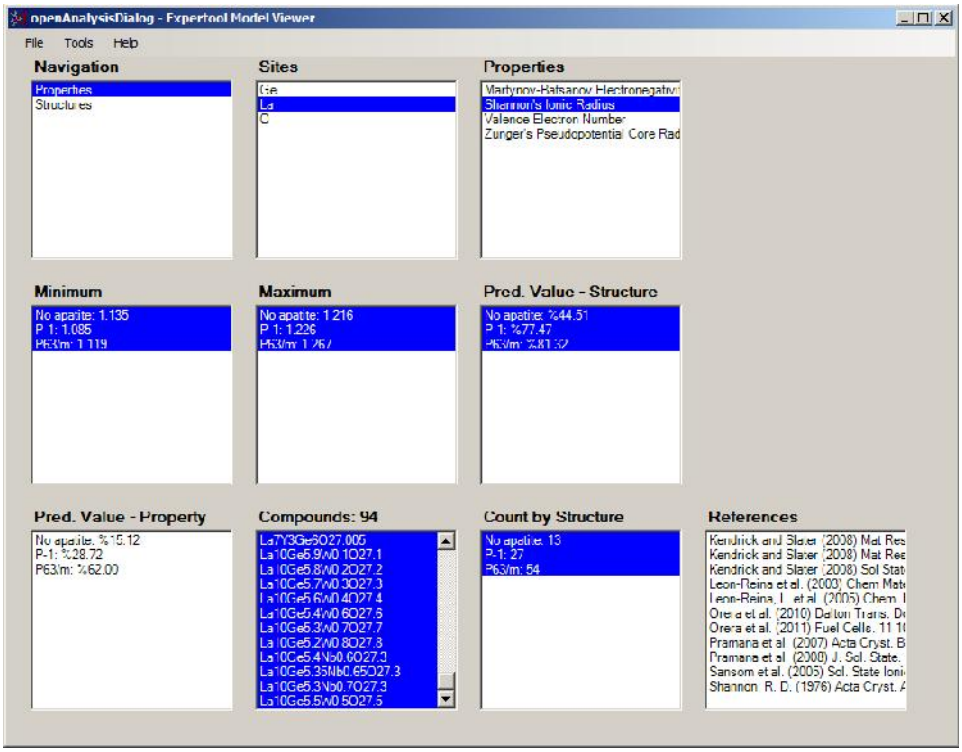
The final quantitative step was to compute for each structure / measurement pair the ratio of the number of compounds with the given structure to the total number of compounds in the entire sample where the specified measurement fell into the range for that structural category. This indicated how consistent a predictor the quantitative measurement was of the structural category.

To facilitate the reading and analysis of the data, the distribution of compounds across results categories with their associated literature references were output to additional windows. The model viewer navigation windows were created and the navigation configuration properties were set to enable the user to browse the results ether of two ways:

- STRUCTURE DRIVEN – A structure is selected, and for each of the eleven measurements the following information is displayed: the maximum and minimum values for the structural category, the maximum and minimum values of the entire sample (for comparison), and the ratio defined in the final quantitative step.



- PROPERTY DRIVEN – A site and a property were selected, and for each structural category the same information was displayed as above. (The maximum and minimum values of the selected measurement for the entire sample were of course only displayed once.)



The "Pred. Value – Structure" window displays the range the property has for the given structure, as a percentage of the overall range of the property across the sample. The smaller the range, the more predictive the structure is of the property.

The "Pred. Value – Property" window displays the number of compounds with the structure, as a percentage of all compounds where the property falls in the range for the structure. The larger the percentage, the more predictive the property is of the structure.

A more advanced analysis that was not included in this proof of concept would have been to select a structural category and multiple measurements, and compute the ratio of the number of compounds in the category to the total within the range on *all* selected measurements. This could have indicated that two or more measurements together are a predictor of structure. A good candidate for this analysis would be where a structure was found to be a good predictor of two or more measurements. No such candidate was present in the sample data used for this model.

The above proof of concept model was scoped to be simple enough to follow, while demonstrating some aspects of how the three key challenges discussed earlier – addressing complexity, computing relevance, capturing and automating human expertise – are being tackled by the Materials and Omics

Modeling Platform research program, using the methods described in this chapter implemented in the Expertool Universal Knowledge Modeling platform.

- Database perspective – the table created as input data is a simple database, but from a methodology perspective represents an array of potential online and offline data sources.
- Ontology perspective – the data is imported into the Expertool neural sandbox structure, making each data point available to behave as a concept or an attribute, depending on how it is contextually categorized by the engine during an analysis.
- Relevance computation perspective – each concept in the ontology has a relevance state (Yes, No or Possible) which is computed at run time by a combination of defined links, available data and user inputs. Each concept has a quantitative state, which is a set of populated and/or computed values which are utilized by the engine as part of the relevance computation.
- Capturing human expertise perspective – in this simple model human expertise was captured first by the use of a data set that was scoped and assembled by cognitive scientists and then by the modeling methods and formulas contributed by the knowledge engineers. This model can now be used as a component of a more comprehensive model or as an agent that interacts with other models.

The scope of the data of this proof of concept is limited, but it can grow without architectural limitations by adding rows to the initial table or adding other tables with related content and defining relationships and interactions that cannot be deduced from the content. The relationships that can be deduced will be identified and updated by the software.

As the scope of the data scales, the combinatorial complexity is attenuated by the use of neural networks to minimize data point and content redundancy and to enable the reuse of links, relationships, formalisms and other components. The Expertool engine computes in stages and at each stage filters for relevance, and only computes those paths found to be relevant, thus minimizing the computing resources required.

As described in the Database section above, in a “big data” scenario, this approach can be used to process a statistically valid sample of the data and identify the relevant paths, which are then passed to the multi-node processing platform (e.g. a Hadoop cluster) to process the available data more efficiently, or in increase the viable scope of data.

The knowledge captured in the model can be dynamically explored in the viewer interface shown in the screen shots above, or accessed by external systems via the API, delivering tailored outputs based on a received set of parameters. The models created by CoSMIC and Expertool will be used to create a dynamic interaction environment for research and discovery using the Models as Agents methodology described earlier.

Conclusion

The prospects associated with materials science and big data are driving high expectations for solutions to perplexing problems and for economic opportunity. However, the realization of these expectations is dependent on creating real value by acquiring knowledge and engineering utility of that knowledge, not by simply accumulating even greater stores of data than that which is overwhelming us today.

To achieve this value informatics practitioners need databases and ontologies. Databases are tools for transforming data into information by putting it in an accessible structure, whereas ontologies are tools for transforming information into knowledge by modeling it in a useful context. Therefore, databases are not platforms for knowledge acquisition and discovery, but rather input sources for knowledge platforms which include ontologies and appropriate engines for their utilization.

Unlike semantic web ontologies that are vocabularies modeled to support search and inference engines, ontologies for scientific research and discovery need to:

- Model the scientific context of the defined concepts
- Model semantic, procedural and episodic abstractions, including quantitative and qualitative elements in a holistic environment, to reflect human knowledge
- Enable chaotic and stochastic interactions at the concept and attribute levels for theory simulation and testing

While the above described approach represents a significant departure from the status quo, it is not based on new theories, but rather on a holistic perspective of the issues and challenges by combining established principles of the computing sciences, cognitive science, complexity theory and engineering disciplines in harmony with the emerging discipline of knowledge engineering.

Further Reading

1. Senge, Peter M. (1990), *The Fifth Discipline: The Art and Practice of the Learning Organization,* Doubleday Currency
2. Schieritz NaM, Peter M (2003). Modeling the forest or Modeling the trees: A comparison of system dynamics and agent-based simulation. *21st International Conference of the System Dynamics Society*, New York.
3. Chomsky N (1957) Syntactic Structures.
4. B. M. Rottman et al, 2011, Causal Systems Categories: Differences in Novice and Expert Categorizations of Causal Phenomena
5. 2008, Understanding Semantic Web Technologies, Ericka Chickowski
6. Spivak DI (2011) Ologs: a categorical framework for knowledge representation.
7. Sica G, ed (2006) What is category theory? Monza, Italy: Polimetrica S.A.S.
8. Alexander, J. H., Freiling, M. J., Shulman, S. J., Staley, J. L., Rehfus, S., and Messick, S. L. 1986 "Knowledge Level Engineering: Ontological Analysis", *Proceedings of AAAI–86*. *Proceedings of the 5th National Conference on Artificial Intelligence*, Los Altos: Morgan Kaufmann Publishers, 963–968.

9. Ashenhurst, Robert L. 1996 “Ontological Aspects of Information Modeling”, *Minds and Machines,* 6, 287–394.
10. Gromiha and Huang BMC Bioinformatics 2012, 13(Suppl 7):I1
11. Rajan, K., ONTOLOGY ENGINEERING: COMPUTATIONAL INFORMATICS FOR AN ICME INFRASTRUCTURE (2011)
12. Peter *The Description Logic Handbook: Theory, Implementation and Applications*, Cambridge: Cambridge University Press:
13. Berners-Lee, Tim, Hendler, James and Lassila, Ora 2001 “The Semantic Web”, *Scientific American*, May 2001.
14. Mayer, R., 1992*, Thinking, Problem Solving, Cognition, Freeman*
15. Reinhold Behringer. "Augmented Reality." In Allen Kent and James G. Williams, eds., *Encyclopedia of Computer Science and Technology*, Vol. 45, No. 30, pp. 45-57. Marcel Dekker, Inc., 2001
16. 2000, Strategy & Business, Between Chaos and Order: What Complexity Theory Can Teach Business
17. Understanding Intelligence, 2001, Pfeifer and Scheier , MIT Press
18. Toshihiko Matsuka et al, Neurocomputing - August 2008
19. Patrick Butler, Ted W. Hall, Alistair M. Hanna, Lenny Mendonca, Byron Auguste, James Manyika, and Anupam Sahay, "A revolution in interaction," The McKinsey Quarterly, 1997 Number 1, pp. 4–23.