Semantics of What?

Werner Kuhn
Institute for Geoinformatics
University of Münster
kuhn@uni-muenster.de
http://musil.uni-muenster.de

Position Paper
NCGIA Specialist Meeting on Spatial Webs
Santa Barbara, CA, December 2-4, 2004

Semantic interoperability is a noble and important goal, but hard to pin down, for several reasons: the notion itself is somewhat redundant, its meaning is elusive as is that of its parts (semantics, interoperability), there is no commonly accepted formal definition, there are no benchmarks and no agreed challenges, the role of humans in the loop is unclear, and the acronym inflation around the semantic web obscures rather than highlights many research issues. As a consequence, semantic interoperability research often doesn't know where to start from, where to go, and how to get there. This position paper tries to determine a starting point of the research by asking *"what is it that needs to be semantically defined in order to support semantic interoperability?"* The intention is to encourage more precise problem statements in semantic interoperability research.

Semantic Interoperability

Before we can determine what it is that needs semantic specifications, we should clarify what we expect to interoperate. In the real world, it is hard to imagine two agents interoperating successfully without a shared understanding of the messages they exchange. If you and a friend prepare a meal without a shared understanding of expressions like "a cup of" or "simmer until tender", no edible food will result. Yet, the kind of interoperability currently supported by standards like those of the Open Geospatial Consortium (OGC) is roughly at this level: interface standards establish syntactic protocols for invoking system behavior, but do not specify in machine-readable form the meaning of the terms used in the protocols. Consequently, OGC tests individual components for conformance to its specifications, but not multiple components for interoperability - whatever that would mean, with or without the "semantic" qualifier. The need to attach semantics to the syntactic protocols has been well recognized in OGC since its early days. Yet, the semantics work of OGC seems to have stalled after establishing its initial grand vision of interoperating information communities and semantic translation.

Interfaces

The early focus of OGC on service interfaces has been very wise. From a semantics point of view, one wishes that this focus had not been lost in the recent shift of attention to GML-based data exchange. But it has: major OGC efforts are now going into database schema harmonization, and the oxymoron of "data interoperability" has crept into industry and agency jargon[1]. It suggests that the flower and eggs in your kitchen interoperate among themselves to prepare your meal. My point is not that humans need to be involved, but that it takes *operations* to interoperate, not just data.

For over 30 years, software engineers have known that data and operation semantics can only be captured together, through interfaces of software modules (Parnas 1972). The idea of a component interface is an ideal (and I would claim the only sound) basis for understanding semantic interoperability. Agents, computational or human, interoperate through interfaces. For them to interoperate in a meaningful way, these interfaces need well-defined semantics. Thus, semantic interoperability is really the only kind of interoperability.

It is essential to consider the human, organizational, and societal issues involved in information sharing and integration. However, it is not helpful to muddle the definition of (technical) interoperability through service interfaces with these aspects. We need clear, crisp, and measurable interoperability criteria if interoperability is to be a research goal. These criteria should be defined incrementally at multiple levels, starting at the technical and proceeding through human, organizational, and social levels.

How can we define interoperability at a technical level? Within and outside the geospatial community, a definition given by ISO TC204 (document N271) is often quoted:

> *"The ability of systems to provide services to and accept services from other systems and to use the services so exchanged to enable them to operate effectively together."*

While this definition is technical enough to be useful in systems engineering and testing, it falls short of establishing verifiable criteria. What does it mean for systems to operate together? And when can they be said to do this effectively? A more precise and verifiable definition of interoperability requires two things:

1. going back to the basics of what software interfaces are and what they mean
2. giving a mathematical definition of interoperability, based on interface specifications.

This paper addresses the first requirement. The second is pursued by people like Joseph Goguen, Michael Grüninger, Robert Kent, Till Mossakowski and Marco Schorlemmer. Preliminary results of an ongoing debate can be found in these colleagues' discussion papers at http://www.dagstuhl.de/04391/Materials/. One can gather from there that the mathematical theory of institutions (Goguen 2004 (draft)), based on category theory, is expected to supply the necessary formal foundations. It rests on *signatures* and establishes the core semantic notion of models satisfying sentences built from signatures in any language. A key question is whether these models should simply be sets (as in classical model theory) or have some algebraic structure imposed by functions.

This paper also does not talk about that other interface: the one between humans and systems. It is equally, if not more important in the quest for semantic interoperability than the machine-level interfaces. The basic point about interface semantics remains the same, but gets complicated by the difficulty to frame human-computer interaction as a formal language, and by the connection to natural language semantics.

---

[1] See, for example, http://www.esri.com/software/arcgis/extensions/datainteroperability/ and http://imgs.intergraph.com/interop/default.asp for industry, and http://www.fgdc.gov/publications/papers/attyszdb.pdf for government.

Signatures

The interface of a software component (such as a web service or a Java package) is formally captured by a *signature*. A signature describes a component's type information, consisting of the input type(s), output type(s) and names of the operations offered by the component. An example of a signature is the specification of distance operations given in the OGC/ISO Spatial Schema standard:

GM_Object :: distance (geometry : GM_Object) : Distance

This signature says that the distance operation is applied to a geometric object (GM_Object), takes as input another such object (called geometry), and returns a value of type Distance.

More generally, the term signature refers to a collection of such operation signatures. Signatures are basic elements of algebra and algebraic software specifications (Ehrig and Mahr 1985). Formally, they contain *three kinds of symbols*, standing for

- functions (of a certain arity)
- sorts (types), and
- constants (tokens).

In the distance example, the only function symbol is **distance**, the sort symbols are **GM_Object** and **Distance**, and there is no symbol for constants. A more mathematical form of the example signature would thus be

distance : **GM_Object** x **GM_Object → Distance**

Note that this form treats the two geometries symmetrically, as it should. A shortcoming of today's software specifications and implementations is that they rarely provide for this decoupling from single object classes.

Semantics

The only sensible use of the term 'semantics' refers to *expressions in a language*. Such expressions can consist of individual symbols ("words") or symbol combinations. Neither concepts nor entities nor properties nor processes have semantics, but the symbols of the languages describing them do (or need).

In an information system context, such languages are manifold: programming languages, schema languages, query languages, interface specification languages, workflow modeling languages, user interface languages, sensor signals, and others. Many of these languages are extensible, i.e., they allow users to introduce new symbols (for data types, attributes, relationships etc.). Additionally, information system standards introduce all sorts of more or less controlled vocabularies (such as those of feature-attribute catalogues or metadata standards). Furthermore, free-form text entries open the gate to almost unlimited uses of natural language expressions in geospatial information sources.

Coping with geospatial semantics means, eventually, building ontologies specifying the meaning of expressions in most or all of these languages. Semantic interoperability research, however, allows us to focus on a very small subset of languages: those defining and using service interfaces. According to the previous section, interfaces are signatures, containing three kinds of symbols as parts of a well-defined structure. Thus, *the semantics required for semantic interoperability is that of expressions built around service signatures*.

Semantics of Signatures

Various languages serve to specify service signatures, to express calls to services or results from them and to reason about services. In the semantic web context, such languages are used to describe, discover, evaluate and invoke web services. For example, WSDL (the Web Service Description Language) allows for syntactic descriptions of web service interfaces and OWL-S (the service ontology of the Web Ontology Language) has been proposed for semantic specifications of services. More comprehensive service modeling efforts like WSMF (the Web Service Modeling Framework) are currently under way (Fensel and Bussler 2002).

Regrettably, the ways in which signatures obtain semantics in these languages are far from clear. Since most ontology languages privilege static entities and binary relationships over more general relationships and processes, it is hard or impossible to say meaningful and useful things about the functions performed by services. The algebraic structure imposed on domains by functions is typically lost. The standard semantic web approach is to specify input and output types, pre- and post-conditions on them, and taxonomies of service types. While input and output specifications can simply point to domain ontologies, pre- and post-conditions require richer models of the domains as well as languages to express rules. The necessary expressiveness and formal bases of these languages are not yet understood, the granularity of service type taxonomies is too coarse, and the service types themselves are not semantically defined.

In other words, while semantic web languages can say something about sets of values or objects in a domain, they are either silent or confused about the mappings between these sets established by services. *Service semantics has no sound formal basis in today's semantic web.* We simply don't know yet what needs to be said (and how) about the semantics of signatures to make the corresponding services semantically interoperable. The suggestions in the semantic web community that a few more half-baked W3C standards will solve these problems are rather optimistic. Much more research is needed on the foundations of service semantics. This research needs to be guided by non-trivial application scenarios.

The case of "spatial webs" offers tremendous application challenges, as well as some solid ground in the form of geospatial information standards. But is spatial information special when it comes to interoperability?

Geospatial Semantics

In the absence of a general theory of service semantics, it is hard to state clearly why and how geospatial services would be special. At the level of establishing semantics for service signatures, there seems not to be anything special about space (and time). Yet, geospatial information and its semantics are characterized by some properties that could guide our search for ways to achieve semantic interoperability. I will close with some guesses about these properties:

1. Geospatial data and services contain symbols whose meaning is not only a matter of convention. For example, a wind direction returned by a weather service, or a water level measured by a gauge have an *observable grounding in the physical world*. Conversely, the meaning of their measuring units, of a currency amount, or of a single-click purchase at Amazon is purely conventional.

2. Because of this physical grounding, explaining the semantics of geospatial information will require elaborate *measurement ontologies*. The emerging sensor web technology will only be useful, if such ontologies are widely available and carefully tied to existing standards in science and engineering.
3. At the same time, geospatial information is often based on *human perception and social agreements*, rather than objective measurements. Coping with the meaning of qualitative judgments (say, of landscape aesthetics) or of social constructions (like neighborhood classifications), and providing mappings among such categories, without imposing incompatible abstractions, are probably the biggest challenges to make geospatial information more meaningful and shareable.
4. A special case of social agreements are *geographic names* and other *identifiers* of geospatial entities. Geographic name registries in the form of gazetteers will need better translation and geo-referencing capabilities. Object identifiers in different databases across information communities will need to be linked. For example, the same petrologic sample could be registered under different identifiers and reference to different geographical names in various online databases supporting geochemical analyses.
5. Space and time are primarily understood through *processes*: we locate stuff because we can move it (not the other way round!), we use distances and directions to navigate, and we determine when to leave the beach by the estimated speed of the advancing storm. This process-nature of geospatial information challenges the entity-bias of the semantic web.
6. Last but not least, *vagueness and different levels of granularity* are fundamental to geospatial information. Theories of vagueness and mappings among granularity levels of geospatial ontologies are therefore essential ingredients of spatial webs.

One approach to these and other requirements is to generalize the notion of spatial reference systems toward *semantic reference systems* (Kuhn 2003). Their task would be to help interpret and translate geospatial information in general, rather than just coordinates. We have only begun to study the computational (Kuhn and Raubal 2003) and institutional challenges posed by this vision. However, pursuing this idea seems a logical consequence of the observation that space itself acts as an integrator of information, through both, locations and the phenomena observed at them.

Conclusions

What is it that needs to be semantically defined in order to support semantic interoperability? This paper has claimed that it is service signatures, which are established notions in mathematics and software engineering. However, it found a lack of understanding how to specify the semantics of signatures in the context of web service description and discovery. And it observed an abandoning, in practice, of service interfaces as the core elements of interoperability, in favor of coarse-grained generic interfaces (like those of map or feature servers) and database schema exchange.

Is a call for fine-grained theories of semantics, at the level of operation signatures, still justified? It may well be that the frustration with fine-grained *syntactic* specifications coming out of the CORBA world was a sign of the need for coarser grained alternatives in semantics as well. On the other hand, we should not throw out the semantic baby with the syntactic bathwater. As long as nobody proposes a mathematical foundation for coarser grained semantic interoperability, we either stick to the small grain, which has such a foundation, or give up on semantic interoperability as a scientific endeavor.

References

Ehrig, H. and B. Mahr (1985). *Fundamentals of Algebraic Specification*. Berlin, Springer-Verlag.

Fensel, D. and C. Bussler (2002). "The Web Service Modeling Framework WSMF." *Electronic Commerce: Research and Applications*(1): 113-137.

Goguen, J. (2004 (draft)). Information Integration in Institutions. (Memorial volume for Jon Barwise). L. Moss. Draft available from http://www.cs.ucsd.edu/~goguen/papers/IF.html.

Kuhn, W. (2003). "Semantic Reference Systems." *International Journal of Geographic Information Science (Guest Editorial)* **17**(5): 405-409.

Kuhn, W. and M. Raubal (2003). Implementing Semantic Reference Systems. AGILE 2003 - 6th AGILE Conference on Geographic Information Science, Lyon, France, Presses Polytechniques et Universitaires Romandes.

Parnas, D. L. (1972). "A Technique for Software Module Specification with Examples." Communications of the ACM **15**(5): 1053-1058.