# SWWS 2001 Position Paper
# Sandro Hawke, W3C

I view the Semantic Web as a practical architecture for a universal information system. It's an old dream: connect the world's computers into one massive system that can take advantage of its reach among locations, its computational resources, and its connections throughout humanity. We've made a lot of progress over the years. I see the Semantic Web as taking several of the next steps, building on successes in different areas, recognizing the principles of open and interoperable systems demonstrated in the history of the Internet and Web.

My interest ranges from the most central peices of the infrastructure up through various application areas. At the low levels, I want to make sure the design is simple enough that a large population can understand it and help it grow. The basic idea of message-passing peer agents presenting a relational database interface seems simple enough; the challenge is to present the real world complications (like network partioning) in the appropriate ways. We need to understand the system well enough to make it look simple.

Moving up the stack, one key to scalable open systems is an open but stable namespace for publishing information, especially meta-information like database schemas, ontologies, rules, and programs. I am very concerned about untangling the complexities of how URI-like-strings are being used as logical symbols and also to identify web pages and content.

Given a simplified relational data model with stable identifiers (symbols), the next step is to add vocabularies for more expressive communication, both for domain-specific areas and for cross-domain fields ranging from the simple (documentation of information) to the more complex (logical formulae defining some terms from others).

The essential components of the Semantic Web, then, are:

1. Layer 1: A language for making simple declarations of fact, using open identifiers which are optionally recognized by various agents. The behavior of agents receiving declarations they do not recognize must be clearly defined and adjustable for different circumstances. This language could be based on the current RDF XML syntax, SQL, KIF, or almost any formal language. It could also be based more directly on arbitrary data formats (eg XML) with a more-complex associated mapping to a relational model. With the right associated language definitions, in fact, we may be able to equivalently use any formal language.

2. Layer 2: Vocabularies for various domains of discourse, allowing Layer 1 declarations to mean something. Everybody should be able to create and disseminate vocabulary terms. Some should be standardized within certain communities for certain purposes, especially the vocabularies for

    - common terms (eg numbers)
    - common information structures (eg sequences)
    - describing vocabularies
    - sets of declarations (information packages)

3. Protocols for exchanging Layer 1 declarations in both active (send or "push") and passive (get) modes. We need protocols which work across slow networks, fast networks, between processes on a computer (possibly running sequentially), and between modules in a process.

4. Finally, as the system evolves, we will need general agent software which can efficiently handle an increasing portion of information handling and processing, following instructions in an evolving vocabulary. This kind of software can be seen as a library or a self-contained agent (or software robot, or daemon), following instructions in a stateless (eg JVM) or stateful (eg DBMS) mannter. As we develop agents which can properly handle this abstracted information processing, more application knowledge will become directly part of the Semantic Web.

All of these technologies already exist in a variety of forms, so perhaps we could say the Semantic Web already exists, but in general they are not interoperable and they are certainly not interoperating on a wide scale. Each of the above essential components needs to be revisited with a clear eye to how it interacts with the other components, across the

wide variety of possible applications.

The most essential applications in my view are (1) the ones which support autocatalysis of the Semantic Web, such as rule-based systems for managing rules and discussion systems for supporting Semantic Web design discussions, and (2) the ones which bi-directionally connect existing information resources (websites, databases) with open and interoperable Semantic Web forms.