Semantic Web Services: Hype or Reality

Ling Liu
College of Computing
Georgia Institute of Technology

Web, Web Services and Semantic Web Services

- Web today:
  - Presents a computing architecture of the Web geared towards delivering information for human-browsing.
- Web Services:
  - Presents a computing architecture of the Web geared towards exchanging information between applications.
- Semantic Web Services:
  - Promises a computing architecture of the Web towards intelligently exchanging information between applications.

My Panel Statement

- Can semantic web services become a reality?
  - Semantic Web Services can play a critical role in the rapid deployment of service oriented applications.
  - However, this will happen only if
    - the use of semantics will not add extra burden on both the developers and the users of Web services;
    - And yet demonstrate real value in terms of productivity.

The History of AI

- Provides many examples of two weaknesses:
  - Over-promising by insiders
    - Intelligent Agents, “a breakthrough in enthusiasm”
  - Over-optimism by outsiders

[Henry S. Thompson, 2000]

- SWS: What should we learn
  - Risk: vulnerable to the same criticism

The History of KR

- The representation and exploitation of knowledge has been the ultimate grand challenge for Artificial Intelligence since its inception
- Human intelligence v.s. machine intelligence
- Lessons learned
  - Designing apparently expressive notations is easy, making them do actual work is *much* harder
  - designing an approach to KR without first designing an inference engine → can be 'a waste of time'
  - User-friendly and Expressive → how to tradeoff

[adopted from Henry S. Thompson 2000]

The Success Story of DBMSs

- Relational data model
  - Mechanisms for capturing simple but powerful semantics (more declarative)
  - Prevailed over Hierarchical data model and Network data model (more procedural)
- RDBMS: ANSI SPARC architecture
  - Three Levels: External, Conceptual, Internal
  - Binding between external and conceptual → Logical data independence
  - Binding between conceptual and internal → Physical data independence

Web Services: an Analogy

- Location-independent distributed computing model
  - Functionality
    - Facilitate rapid design and deployment of service oriented applications
    - Enabling effective creation, execution, and composition, as well as automated discovery and classification, of Web services
- Productivity and Ease of Deployment
  - Three tier architecture
    - User Level, Service Level, Execution Level
    - Binding between user and service → **Service Location Independence**
    - Binding between service and execution → **Service Execution** (interface/invocation) **Independence**

Web Services: On the Move

- **Web Services as a Software Architecture**
  - connect computers and devices with each other through the Web infrastructure and exchange and combine Web data dynamically
  - enable software to be delivered as continuous streams of services as opposed to packaged products.
- **Web Services as a new Concept for enterprise application integration**
  - The integration of data, information, knowledge; processes; applications; and business
- **Web Services as a Web programming technology**
  - **today:** Remote Procedure Calls (RPC) over HTTP, powered by WSDL, UDDI, SOAP, BPEL etc.
  - **Tomorrow**: location-independent distributed computational model, powered by three tier distributed service computing architecture

Take Home Message:

- Research and Standardization should focus on methodology and technologies that encourage and enable *the use of semantics to* ***increase the productivity*** *of both the developers and the users of Web services and applications*