Outline of Talk

• Semantic Web overview
• Web Services overview
• Semantic Web Services overview
• Semantic Web Rules
  – RuleML
  – Uses in Semantic Web Services
  – Example: SweetDeal e-contracting
• Early Adopter Areas
  – Discussion
The Semantic Web

The 1st generation, the Internet, enabled disparate machines to exchange data.

• The 2nd generation, the World Wide Web, enabled new applications on top of the growing Internet, making enormous amounts of information available, in human-readable form, and allowing a revolution in new applications, environments, and B2C e-commerce.

• The next generation of the net is an “agent-enabled” resource (the “Semantic Web”) which makes a huge amount of information available in machine-readable form creating a revolution in new applications, environments, and B2B e-commerce.

…by enabling “agent” communication at a Web-wide scale.
Web is becoming XML → the **Semantic Web**

- XML (vs. HTML) offers much greater capabilities for structured detailed descriptions that can be processed automatically.
  - Eases application development effort for assimilation of data in inter-enterprise interchange
  - A suite of open standards both current and emerging
  - … including for knowledge-level SEMANTICS
- Soon, Agents will Talk according to these standards…
  - .::. potential to revolutionize interactivity in Web marketplaces
- B2B, …
- HTML itself is becoming XHTML: just a special case of XML
Vision of Evolution: Agents in Knowledge-Based E-Markets

Coming soon to a world near you:...

- billions/trillions of agents (= k-b applications)
- ...with smarts: knowledge gathering, reasoning, economic optimization
- ...doing our bidding
  - but with some autonomy
- A 1st step: ability to communicate with sufficiently precise shared meaning... via the SEMANTIC WEB
SW: Research Players

• US: DARPA Agent Markup Language Program (DAML) program
• EU: OntoWeb program
• @MIT:
  – Sloan IT group: Grosof, Madnick, Firat, Klein, et al
  – LCS / W3C advanced-dev.: Berners-Lee, et al
• Number of companies:
  – HP, IBM, Adobe, Oracle, …
Semantic Web “Stack”: Standardization Steps

Emerging Standards pioneered in DARPA Agent Markup Language (DAML) program: e.g.

- RuleML
- OWL/DAML+OIL

[Diagram http://www.w3.org/DesignIssues/diagrams/sw-stack-2002.png is courtesy Tim Berners-Lee]
**SW Stack: Acronym Expansion**

- **W3C** = World Wide Web Consortium: umbrella standards body
- **XML-S**: XML Schema, i.e., basic XML spec
- **RDF**: Resource Description Framework:
  - W3C Working Group
  - Labelled directed graph syntax
  - Good for building knowledge representation on top of: simpler, more powerful than basic XML
  - M&S = Model and Syntax
  - RDF Schema = extension: simple class hierarchies
- **Ontology** = formally defined vocabulary & class hierarchy, generalizes Entity-Relationship models
  - OWL = W3C Web Ontologies Working Language
  - … based closely on DAML+OIL
SW: Standards Players

- US-EU Joint Committee:
  - Early standards drafting
  - 1st focus: ontologies: DAML+OIL → W3C OWL
  - 2nd focus (current): rules: RuleML

- W3C: Semantic Web Activity

- Oasis: various incl. Security

- New efforts (currently in formation):
  - US-EU Joint Committee on Semantic Web Services
  - ISO: CommonLogic first-order logic (formerly KIF)
SW-Related: XML Query Languages

• Goals
  – a data model for generic “natively” XML documents,
  – a set of query operators on that data model,
  – and a query language based on these query operators
  – Queries operate on single documents or fixed collections of documents.

• What SQL is for relational databases, XML Query languages are for collections of XML docs.

• There is a standard: W3C’s XML Query Working Group
  – (W3C = World Wide Web Consortium)

• Oracle, IBM, Microsoft, etc. already support some
  – Not taking off quickly – complex spec
Outline of Talk

• Semantic Web overview
• Web Services overview
• Semantic Web Services overview
• Semantic Web Rules
  – RuleML
  – Uses in Semantic Web Services
  – Example: SweetDeal e-contracting
• Early Adopter Areas
  – Discussion
Web Service -- definition

• *(For purposes of this talk:)*

• A procedure/method that is invoked through a Web protocol interface, typically with XML inputs and outputs
Current Web Services Standards Stack; Context for Semantic Web Services

Wire Protocols
- SOAP Blocks
- SOAP/XMLP
- XML
- HTTP/SMTP/BEEP
- TCP/IP

Description
- W3C WS Choreography Group
- BPEL4WS (Microsoft, IBM, BEA)
- WSCL (HP)BPML (Most but Microsoft)
- WSCI (Sun, BEA, Yahoo, …)
- XLANG (Microsoft), WSFL (IBM), …
- Agreements
- Process
- WSDL Extensions
- WSDL
- XML

Automated
- Discovery
- Invocation
- Interoperation
- Deal Negotiation
- Composition
- Monitoring
- Verification

DAML
- Registry (UDDI)
- Inspection

[Modification of slide by James Snell (IBM)]
WS Stack: some Acronym Expansion

- SOAP = simple protocol for XML messaging
- WSDL = protocol for basic invocation of Web Services, their input and output types in XML
- Choreography = higher-level application interaction protocols in terms of sequences of exchanged message types, contingent branching
  - Currently morphing into a W3C activity

Overall: lots of proprietary jockeying and de-facto mode testing/pressuring of the open-consortial standards bodies (e.g., of W3C) “riding the tiger”
WS Players

• Basically, all the major software vendors
  – Biggies: Microsoft, IBM, Oracle, Sun, SAP, …
  – Webserver/XML ebiz space: BEA, CommerceOne, Ariba, …
  – Niche offerings, e.g., travel agent services, weather, …
• Standards bodies: W3C; Oasis incl. Security
• Overall: lots of proprietary jockeying and de-facto mode testing/pressuring of the open-consortial standards bodies (e.g., of W3C) “riding the tiger”
• Still low-level in terms of application abstractions
Outline of Talk

- Semantic Web overview
- Web Services overview
- Semantic Web Services overview
- Semantic Web Rules
  - RuleML
  - Uses in Semantic Web Services
  - Example: SweetDeal e-contracting
- Early Adopter Areas
  - Discussion
Semantic Web Services

• Convergence of Semantic Web and Web Services
• Consensus definition and conceptualization still forming
• Semantic (Web Services):
  – Knowledge-based service descriptions, deals
    • Discovery/search, invocation, negotiation, selection, composition, execution, monitoring, verification
  – Integrated knowledge
• (Semantic Web) Services: e.g., infrastructural
  – Knowledge/info/DB integration
  – Inferencing and translation
Current Web Services Standards Stack; Context for Semantic Web Services

Wire Protocols
- SOAP Blocks
- SOAP/XML
- XML
- HTTP/SMTP/BEEP
- TCP/IP

Description
- W3C WS Choreography Group
- BPEL4WS (Microsoft, IBM, BEA)
- WSCL (HP)BPML (Most but Microsoft)
- WSCI (Sun, BEA, Yahoo, …)
- XLANG (Microsoft), WSFL (IBM), …

Agreements
- Process
- WSDL Extensions
- WSDL
- XML

Automated Discovery
- Invocation
- Interoperation
- Composition
- Monitoring
- Verification

DAML
- Registry (UDDI)
- Inspection

[Modification of slide also by Sheila McIlraith (Stanford) and David Martin (SRI International), modified from James Snell (IBM)]
**SWS Tasks at higher layers of WS stack**

Automation of:

- **Web service discovery**
  
  *Find me a shipping service that will transport frozen vegetables from San Francisco to Tuktoyuktuk.*

- **Web service invocation**
  
  *Buy me “Harry Potter and the Philosopher’s Stone” at www.amazon.com*

- **Web service deals, i.e., contracts, and their negotiation**
  
  *Propose a price with shipping details for used Dell laptops to Sue Smith.*

- **Web service selection, composition and interoperation**
  
  *Make the travel arrangements for my WWW11 conference.*

[Modification of slide also by Sheila McIlraith (Stanford) and David Martin (SRI International)]

11/22/2002 Copyright 2002 by Benjamin Grosof. All Rights Reserved
SWS Tasks at higher layers of WS stack, continued

- Web service **execution monitoring** and **problem resolution**
  
  Has my book been shipped yet? ... [NO!] Obtain recourse.

- Web service **simulation** and **verification**
  
  Suppose we had to cancel the order after 2 days?

- Web service **executably specified at “knowledge level”**
  
  The service is performed by running the contract ruleset through a rule engine.

[Modification of slide also by Sheila McIlraith (Stanford) and David Martin (SRI International)]
Vision: Semantic Web and Web Services
Use DB’s, Ontologies, and Rule Systems

Rules good for contingent aspects of service descriptions

Services: DAML-S, WSMF

Rules: RuleML

Ontologies: OWL

Databases: SQL, XQuery, RDF
SWS: Research Players

• DAML Services (DAML-S)
  – service descriptions using ontologies and now rules
• Web Services Mediator Framework (WSMF)
  – EU, Oracle
  – early phase; list of many companies
• @ MIT: Sloan IT:
  – SweetDeal: e-contracting, policies
  – Extended COIN: financial info integration
Outline of Talk

• Semantic Web overview
• Web Services overview
• Semantic Web Services overview
• Semantic Web Rules
  – RuleML
  – Uses in Semantic Web Services
  – Example: SweetDeal e-contracting
• Early Adopter Areas
  – Discussion
Flavors of Rules Commercially Most Important today in E-Business

• E.g., in OO app’s, DB’s, workflows.

• Relational databases, SQL: Views, queries, facts are all rules.
  • SQL99 even has recursive rules.

• Production rules (OPS5 heritage): e.g.,
  – Blaze, ILOG, Haley: rule-based Java/C++ objects.

• Event-Condition-Action rules (loose family), cf.:
  – business process automation / workflow tools.
  – active databases; publish-subscribe.

• Prolog. “logic programs” as a full programming language.

• (Lesser: other knowledge-based systems.)
Vision: Uses of Rules in E-Business

• Rules as an important aspect of coming world of Internet e-business: rule-based business policies & business processes, for B2B & B2C.
  – represent seller’s offerings of products & services, capabilities, bids; map offerings from multiple suppliers to common catalog.
  – represent buyer’s requests, interests, bids; → matchmaking.
  – represent sales help, customer help, procurement, authorization/trust, brokering, workflow.
  – high level of conceptual abstraction; easier for non-programmers to understand, specify, dynamically modify & merge.
  – executable but can treat as data, separate from code
    • potentially ubiquitous; already wide: e.g., SQL views, queries.
• Rules in communicating applications, e.g., embedded intelligent agents.
Why Standardize Rules Now?

- Rules as a form of KR (knowledge representation) are especially useful:
  - relatively mature from basic research viewpoint
  - good for prescriptive specifications (vs. descriptive)
    - a restricted programming mechanism
  - integrate well into commercially mainstream software engineering, e.g., OO and DB
    - easily embeddable; familiar
    - vendors interested already: Webizing, app. dev. tools
- ⇒⇒ Identified as part of mission of the W3C Semantic Web Activity
Overview of RuleML Today

• RuleML Initiative (2000--)
  – Dozens of institutions (~35), researchers; esp. in US, EU
  – Mission: Enable semantic exchange of rules/facts between most commercially important rule systems
  – Standards specification: 1st version 2001; basic now fairly stable
  – A number of tools (~12 engines, translators, editors), demo applications
  – Successful Workshop on Rules at ISWC was mostly about RuleML / LP
  – Has now a “home” institutionally in DAML and Joint Committee
    • Discussions well underway to launch W3C, Oasis efforts

• Initial Core: Horn Logic Programs KR
  …Webized (in markup)… and with expressive extensions

  URI’s, XML, RDF, …    non-mon, actions, …
Overview of RuleML Today, Continued

• Fully Declarative KR (not simply Prolog!)
  – Well-established logic with model theory
  – Available algorithms, implementations
  – Close connection to relational DB’s; core SQL is Horn LP
  – See [Baral & Gelfond ’94] for good survey on declarative LP.

• Abstract graph syntax
  – 1st encoded in XML…
  – … then RDF (draft), … then DAML+OIL (draft)

• Expressive Extensions incrementally, esp. already:
  – Non-monotonicity: Negation as failure; Courteous priorities
  – Procedural Attachments: Situated actions/effecting, tests/sensing
  – In-progress: Events cf. OPS5/Event-Condition-Action
Rule-based Semantic Web Services

- Rules/LP in appropriate combination with DL as KR, for RSWS
  - DL good for **categorizing**: a service overall, its inputs, its outputs

- Rules to describe **service process models**
  - rules good for representing:
    - **preconditions** and **postconditions**, their contingent relationships
    - **contingent** behavior/features of the service more generally,
      - e.g., exceptions/problems
    - familiarity and naturalness of rules to software/knowledge engineers

- Rules to specify **deals about services**: cf. e-contracting.
Rule-based Semantic Web Services

• Rules often good to **executably specify** service process models
  – e.g., business process automation using procedural attachments to perform side-effectful/state-changing actions ("effectors" triggered by drawing of conclusions)
  – e.g., rules obtain info via procedural attachments ("sensors" test rule conditions)
  – e.g., rules for knowledge translation or inferencing
  – e.g., info services exposing relational DBs

• **Infrastructural**: rule system functionality as services:
  – e.g., inferencing, translation
Application Scenarios for Rule-based Semantic Web Services

• SweetDeal [Grosof & Poon 2002] configurable reusable e-contracts:
  – LP rules about agent contracts with exception handling
  – … on top of DL ontologies about business processes;
  – a scenario motivating DLP

• Other:
  – Trust management / authorization (Delegation Logic) [Li, Grosof, & Feigenbaum 2000]
  – Financial knowledge integration (ECOIN) [Firat, Madnick, & Grosof 2002]
  – Privacy policies (P3P APPEL)
  – Business policies, more generally
Slides on SweetDeal: Pointer

- See talk slides (from ISWC Rules Workshop 2002) at http://ebusiness.mit.edu/bgrosof/#SweetDealExceptions

- Next few slides, taken from that, give a sample.
Contract Rules during Negotiation

Contracting parties NEGOTIATE via shared rules.
Overview I: SweetDeal, Exception Handlers, Web Services

- This work is part of SweetDeal: rule-based approach for e-contracting
- Advantages of rule-based: (use Situated Courteous LP KR in RuleML)
  - high level of conceptual abstraction to specify;
  - modularly modifiable; reusable; executable
  - esp. good for specifying contingent provisions
- Reusable ruleset modules represent parts of contracts
- Here, newly extend to include exception handlers:
  - violations of commitments → invoke business processes
  - more complex behavior
  - good for services, e.g., deals about Web services
- process descriptions whose ontologies are in DAML+OIL
  - drawn from MIT Process Handbook, a previous repository
    - uniquely large & well-used (by industry biz process designers)
  - partially or fully specified by rules (executably)
Example Contract Proposal with Exception Handling
Represented using RuleML & DAML+OIL, Process Descriptions

buyer(co123,acme);
seller(co123,plastics_etc);
product(co123,plastic425);
price(co123,50);
quantity(co123,100);
http://xmlcontracting.org/sd.daml#Contract(co123);
http://xmlcontracting.org/sd.daml#specFor(co123,co123_process);
http://xmlcontracting.org/sd.daml#BuyWithBilateralNegotiation(co123_process);
http://xmlcontracting.org/sd.daml#result(co123,co123_res);
shippingDate(co123,3); // i.e. 3 days after order placed
// base payment = price * quantity
payment(?R,base,?Payment) <-
  http://xmlcontracting.org/sd.daml#result(co123,?R) AND
  price(co123,?P) AND quantity(co123,?Q) AND
  multiply(?P,?Q,?Payment) ;

Using concise text syntax
(SCLP textfile format)
for concise human reading
SCLP TextFile Format for (Daml)RuleML

\[\text{payment}(\text{?R}, \text{base}, \text{?Payment}) \leftarrow \]
\[\text{http://xmlcontracting.org/sd.daml#result}(\text{co123}, \text{?R}) \text{ AND}
\text{price}(\text{co123}, \text{?P}) \text{ AND quantity}(\text{co123}, \text{?Q}) \text{ AND}
\text{multiply}(\text{?P}, \text{?Q}, \text{?Payment}) ;\]

\[<\text{drm:imp}>
   <\text{_head}> <\text{drm:atom}>
      <\text{_opr}><\text{rel}>\text{payment}</\text{rel}></\text{_opr>}
      <\text{tup>}
         <\text{var}>R</\text{var> <\text{ind}>base</\text{ind> <\text{var}>Payment</\text{var>}
      </\text{tup>}></\text{atom>}
   </\text{_head>}
   <\text{_body>}
      <\text{andb>}
         <\text{atom>}
            <\text{_opr><\text{rel>href="http://xmlcontracting.org/sd.daml#result"/>}
            <\text{_opr>}
               <\text{tup>}
                  <\text{ind}>co123</\text{ind> <\text{var}>Cust</\text{var>}
                  </\text{tup>}
         </\text{atom>}
      </\text{andb>}
   </\text{_body>}
</\text{drm:imp>}

drm = namespace for damlRuleML
lateDeliveryPenalty_module {
// lateDeliveryPenalty is an instance of PenalizeForContingency
// (and thus of AvoidException, ExceptionHandler, and Process)
http://xmlcontracting.org/pr.daml#PenalizeForContingency(lateDeliveryPenalty) ;
// lateDeliveryPenalty is intended to avoid exceptions of class
// LateDelivery.
http://xmlcontracting.org/sd.daml#avoidsException(lateDeliveryPenalty,
   http://xmlcontracting.org/pr.daml#LateDelivery);
// penalty = - overdueDays * 200 ; (negative payment by buyer)

<lateDeliveryPenalty_def> payment(?R, contingentPenalty, ?Penalty) <-
   http://xmlcontracting.org/sd.daml#specFor(?CO, ?PI) AND
   http://xmlcontracting.org/pr.daml#hasException(?PI, ?EI) AND
   http://xmlcontracting.org/pr.daml#isHandledBy(?EI, lateDeliveryPenalty) AND
   http://xmlcontracting.org/sd.daml#result(?CO, ?R) AND
   http://xmlcontracting.org/sd.daml#exceptionOccurred(?R, ?EI) AND
   shippingDate(?CO, ?CODate) AND shippingDate(?R, ?RDate) AND
   subtract(?RDate, ?CODate, ?OverdueDays) AND
   multiply(?OverdueDays, 200, ?Res1) AND multiply(?Res1, -1, ?Penalty) ;
}
<lateDeliveryPenaltyHandlesIt(e1)> // specify lateDeliveryPenalty as a handler for e1
http://xmlcontracting.org/pr.daml#isHandledBy(e1, lateDeliveryPenalty);
Outline of Talk

• Semantic Web overview
• Web Services overview
• Semantic Web Services overview
• Semantic Web Rules
  – RuleML
  – Uses in Semantic Web Services
  – Example: SweetDeal e-contracting
• Early Adopter Areas
  – Discussion
SW Early Adoption Candidates: High-Level View

• “Death. Taxes. Integration.”

• Application/Info Integration:
  – Intra-enterprise
    • EAI, M&A; XML infrastructure trend
  – Inter-enterprise
    • E-Commerce: procurement, SCM
  – Combo
    • Business partners, extranet trend
SW Early Adopters: 
Areas by Industry or Task

- Early SW techniques already in use:
  - e-contracting, supply chain incl. procurement
    - manufacturing, e.g. computer/electronics (RosettaNet), automotive (Covisint),
    - EECOMS pilot (Boeing, IBM, TRW, Baan)
  - office supplies (OBI)
  - retailing: shopbots and salesbots: comparisons, recommendations
  - extensive standards activity: Oasis ebXML, XML eContracts, UN UBL, EDI
SW Early Adopters: Areas by Industry or Task

• Continued: Early SW techniques already in use:
  – cyber goods:
    • financial services (rules; onto translation)
    • travel "agency", i.e.: tickets, packages (AI smarts for scheduling)
  – military intelligence (e.g., funded DAML)
SW Early Adopters:  
Areas by Industry or Task

• Still in research or early standardization, mainly:
  – e-contracting:
    • auctions
    • construction
    • insurance, risk management
    • SME's, spontaneity
    • international
    • distribution
  – authorization and security policies
  – business policies
SW Early Adopters: Areas by Industry or Task

• Continued: Still in research or early standardization, mainly:
  – reputations, ratings
  – legal/regulatory: forms, dispute resolution; Oasis Legal XML
  – computer games: massive multi-player
  – question-answering
  – news filtering, e.g., financial
  – knowledge management
  – advertising
  – bioinformatics, scientific Grid
FOR MORE INFO -- on author’s webpage

• At http://ebusiness.mit.edu/bgrosof:
    • …/#SweetDealExceptions
  – RuleML Overviews
    • …/#RuleML, esp. 10/29/02 Joint Committee intro talk
  – Description Logic Programs paper and talk (discusses deeper technical approach to combining rules and ontologies)
    • …/#DLP
  – SWS Project overviews
    • …/#Overview and …/#Projects
FOR MORE INFO - resources on SW, WS, SWS

- SWS overview: http://ebusiness.mit.edu/#SWS
- DAML http://www.daml.org; esp. DAML-S .../services
- WSMF http://informatik.uibk.ac.at/users/c70385/wese/publications.html
- W3C SW: http://www.w3.org/2001/sw -> charter, RDF, WebOnt
- Also at W3C: WSDL, Xquery, ...
- Web Services – Interoperability http://www.ws-i.org
- Oasis XML standards body http://www.oasis-open.org
- RuleML main site (major editing in progress): http://www.ruleml.org
- And:
  - XML world: the Cover pages http://xml.coverpages.org
  - A SW community portal http://www.semanticweb.org