E-Services on the New Generation Web: Automating Business Process Knowledge Management

Benjamin Grosof
MIT Sloan School of Management
Information Technologies group
http://ebusiness.mit.edu/bgrosof

Slides presented at Center for eBusiness @ MIT Research Seminar,
Apr. 14, 2004
MIT Sloan School of Management
http://ebusiness.mit.edu
Outline of Talk

• Intro: Research on Semantic Web Services (SWS), its Business Uses
  – Rules, contracting, trust, policies
  – Integration, knowledge representation, standards
• Problem: Reusable Knowledge to Describe Services
  – Technique: knowledge representation to standardize on
  – Content investment: how to leverage legacy business process K
• New Technical Approach to represent OO Frameworks using SW
  – Courteous Inheritance: default rules increases reuse in ontologies
• New Strategy: go where the knowledge already is, then work outwards
  – Begin with MIT Process Handbook – open-source version in development
    • Example: process knowledge about selling
  – Future: Transformational wrappers around various legacy OO frameworks
• Roadmapping Market Evolution
  – Early adopters, creators, catalysts
  – Strategic players, forces
Next Generation Web

Semantic Web Services

Semantic Web techniques

Web Services techniques

Automated Knowledge Bases

API’s on Web

Two interwoven aspects:
Program: Web Services
Data: Semantic Web

First Generation Web

XML
Brief Tour of some relevant websites

- http://www.oasis-open.org Oasis, e.g., its web services standards
Big Questions about the New Generation Web

- What are the critical features/aspects of the new technology?
- What business problems does it help solve?
- What are the likely innovation evolution paths, and associated entrepreneurial opportunities?
Our Overall SWS Research Agenda

- Invent Core Technologies and concepts of the New Generation Web
  - Semantic Web; Rules and RuleML emerging standard
    • supporting knowledge representation theory of Situated Courteous Description Logic Programs
  - Semantic Web Services; Business Process Automation for B2B and EAI
    • Requirements analysis

- Pilot Business Application Scenarios
  - End-to-end e-contracting, e.g., in manufacturing supply chain
    • SweetDeal approach using rules
  - Financial information and reporting:
    • ECOIN approach mapping ontologies
  - Other: security authorization, travel, …

- Analyze Prospective Early Adopter Areas
  - Strategy: Adoption Roadmap; Market Evolution
  - Entrepreneurial Opportunities
Some Answers to: “Why does SWS Matter to Business?”


2. “Business processes require communication between organizations / applications.” - Data and programs cross org./app. boundaries, both intra- and inter- enterprise.

3. “It’s the automated knowledge economy, stupid!” - The world is moving towards a knowledge economy. And it’s moving towards deeper and broader automation of business processes. The first step is automating the use of structured knowledge.
   - Theme: *reuse* of knowledge across multiple tasks/app’s/org’s
B2B Tasks: Communication for Business Processes with Partners

- B2B business processes involving significant Communication with customers/suppliers/other-partners is overall a natural locus for future first impact of SWS.
- Customer Relationship Management (CRM):
  - sales leads and status
  - customer service info and support
- Supply Chain Management (SCM):
  - source selection
  - inventories and forecasts
  - problem resolution
  - transportation and shipping, distribution and logistics
- orders; payments, bill presentation
Some B2B Tasks (continued)

- bids, quotes, pricing, CONTRACTING; AUCTIONS; procurement
- authorization (vs. authentication) for credit or trust
- database-y: e.g.,
  - catalogs & their merging
  - policies
- inquiries and answers; live feedback
- notifications
- trails of biz processes and interactions
- ratings, 3rd party reviews, recommendations
- knowledge management with partners/mkt/society
New Research Application Scenarios for Rule-based Semantic Web Services

- **SweetDeal** [Grosof & Poon WWW-2003] configurable reusable e-contracts:
  - Represents modular modification of proposals, service provisions
  - LP rules as KR. E.g., prices, late delivery exception handling.
  - On top of DL ontologies about business processes from MIT Process Handbook
  - Evolved from EECOMS pilot on agent-based manufacturing SCM ($51M NIST ATP 1996-2000 IBM, Boeing, TRW, Vitria, others)

- **Financial** knowledge integration (ECOIN) [Firat, Madnick, & Grosof 2002]
  - Maps between contexts using LP rules, equational ontologies, SQL DB’s.

- **Business Policies**:
  - Trust management (Delegation Logic) [Li, Grosof, & Feigenbaum 2003]:
OPTIONAL BACKUP SLIDES
FOLLOW

- About what are Semantic Web, Web Services, and Semantic Web Services
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
Web Services Stack outline

NOTES:

WSDL is a Modular Interface spec
SOAP is Messaging and Runtime
Also:
- UDDI is for Discovery
- BPEL4WS, WSCI, … are for transactions
- Routing, concurrency, …

Diagram courtesy Tim Berners-Lee: http://www.w3.org/2004/Talks/0309-ws-sw-tbl/slide6-0.html
Semantic Web: concept, approach, pieces

- Shared semantics when interchange data ⊨ knowledge
- **Knowledge Representation** (cf. AI, DB) as approach to semantics
  - Standardize KR syntax, with KR theory/techniques as backing
- Web-exposed **Databases**: SQL; XQuery (XML-data DB’s)
  - Challenge: share DB schemas via meta-data
- **RDF**: “Resource Description Framework” W3C proposed standard
  - Meta-data lower-level mechanics: unordered directed graphs (vs. ordered trees)
  - **RDF-Schema** extension: simple class/property hierarchy, domains/ranges
- **Ontology** = formally defined vocabulary & class hierarchy
  - **OWL**: “Ontologies Working Language” W3C proposed standard
    - Subsumes RDF-Schema and Entity-Relationship models
    - Based on Description Logic (DL) KR ~subset of First-Order Logic (FOL))
- **Rules** = if-then logical implications, facts ~subsumes SQL DB’s
  - **RuleML**: “Rule Markup Language” emerging standard
    - Based on Logic Programs (LP) KR ~extension of Horn FOL
W3C Semantic Web “Stack”: Standardization Steps

Emerging Standards
pioneered in DARPA Agent Markup Language (DAML) program:
• RuleML
• OWL

[Diagram http://www.w3.org/DesignIssues/diagrams/sw-stack-2002.png is courtesy Tim Berners-Lee]
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
END OF
OPTIONAL BACKUP SLIDES

• About what are Semantic Web, Web Services, and Semantic Web Services
Outline of Talk

• Intro: Research on Semantic Web Services (SWS), its Business Uses
  – Rules, contracting, trust, policies
  – Integration, knowledge representation, standards
• Problem: Reusable Knowledge to Describe Services
  – Technique: knowledge representation to standardize on
  – Content investment: how to leverage legacy business process K
• New Technical Approach to represent OO Frameworks using SW
  – Courteous Inheritance: default rules increases reuse in ontologies
• New Strategy: go where the knowledge already is, then work outwards
  – Begin with MIT Process Handbook – open-source version in development
    • Example: process knowledge about selling
  – Future: Transformational wrappers around various legacy OO frameworks
• Roadmapping Market Evolution
  – Early adopters, creators, catalysts
  – Strategic players, forces
Problem: Reusable Knowledge to Describe Services

- Has two aspects:

1. **Technical/technique problem**: what form of knowledge? I.e., what knowledge representation to standardize on?

2. **Content investment problem**: how to leverage to accomplish the reuse of legacy business process knowledge?
Outline of Talk

• Intro: Research on Semantic Web Services (SWS), its Business Uses
  – Rules, contracting, trust, policies
  – Integration, knowledge representation, standards
• Problem: Reusable Knowledge to Describe Services
  – Technique: knowledge representation to standardize on
  – Content investment: how to leverage legacy business process K
• New Technical Approach to represent OO Frameworks using SW
  – Courteous Inheritance: default rules increases reuse in ontologies
• New Strategy: go where the knowledge already is, then work outwards
  – Begin with MIT Process Handbook – open-source version in development
    • Example: process knowledge about selling
  – Future: Transformational wrappers around various legacy OO frameworks
• Roadmapping Market Evolution
  – Early adopters, creators, catalysts
  – Strategic players, forces
Opportunity for MIT Process Handbook in SWS

• Need for Shared Web Services / Business Processes Knowledge Bases

• MIT Process Handbook as candidate nucleus for shared business process ontology for SWS
  – 5000+ business processes, + associated class/property concepts, as structured knowledge

• Related: use in particular for E-Contracting
  – Interoperable business objects, business processes
  – Also for policies (e.g., trust), 3rd-party services
Some Specializations of “Sell” in the MIT Process Handbook (PH)
OPTIONAL BACKUP SLIDES FOLLOW

• About SweetDeal’s use of Process Handbook ontology in rule-based e-contracts
Some Exceptions in the MIT Process Handbook
Some exception handlers in the MIT Process Handbook

- Anticipate exception
  - Detect fraudulent reputation
  - Maintain reputation information
  - Track kBT
  - Determine maximum resources
  - Determine behavior outside normal

- Avoid exception
  - Detect via notification
  - Detect prerequisite violation
  - Detect poor demand forecast
  - Detect skill bidding
  - Detect timeout
  - Poll
  - Monitor using sentinels

- Detect exception
  - Detect protocol violation
  - Observe many low priority tasks
  - Compare priorities of current and previous
  - Detect agent jumping a lot
  - Require response from subordinate
  - Determine lack of response
  - Detect too many bid collisions
  - Receive message from 'dead' a...

- Manage exception
  - Notify about exception using pager
  - Notify about exception using mail
  - Notify about exception using email

- Resolve exception
  - Pre-emptive reallocation
  - Terminate bid loop, auctioneer aw...
  - Reset price and restart auction
  - Negotiate
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);

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

  </drm:_body> </drm:imp>
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) ;
• About SweetDeal’s use of Process Handbook ontology in rule-based e-contracts
Outline of Talk

• Intro: Research on Semantic Web Services (SWS), its Business Uses
  – Rules, contracting, trust, policies
  – Integration, knowledge representation, standards

• Problem: Reusable Knowledge to Describe Services
  – Technique: knowledge representation to standardize on
  – Content investment: how to leverage legacy business process K

• New Technical Approach to represent OO Frameworks using SW
  – Courteous Inheritance: default rules increases reuse in ontologies

• New Strategy: go where the knowledge already is, then work outwards
  – Begin with MIT Process Handbook – open-source version in development
    • Example: process knowledge about selling
  – Future: Transformational wrappers around various legacy OO frameworks

• Roadmapping Market Evolution
  – Early adopters, creators, catalysts
  – Strategic players, forces

- Use SW KR and standards to represent Object-Oriented framework knowledge: class hierarchy, types, generalization-specialization, domain & range, properties/methods’ association with classes
- Surprise: use SW rule language not the main SW ontology language! I.e., use RuleML not OWL.
- Exploit RuleML’s nonmonotonic ability to represent prioritized default reasoning as kind of knowledge representation (KR)
New Technical Approach, continued

• Courteous Inheritance KR is built simply on top of the (Situated) Courteous Logic Programs KR of RuleML
  – A few dozen background axioms. Linear-size reformulation. Inferencing is tractable computationally.
• Particularly: represent PH's structured part
  – a scheme specific to PH’s flavor of OO
• PH becomes a SWS process ontology
New Technical Approach, continued more

• Example(s): selling, PO, price, shipping, delivery, payment, lateness.

• For details, see submitted paper “Beyond Monotonic Inheritance: Towards Semantic Web Process Ontologies” on webpage.
  – Example: selling process
Brief Tour of selling example in the paper.
Outline of Talk

• Intro: Research on Semantic Web Services (SWS), its Business Uses
  – Rules, contracting, trust, policies
  – Integration, knowledge representation, standards

• Problem: Reusable Knowledge to Describe Services
  – Technique: knowledge representation to standardize on
  – Content investment: how to leverage legacy business process K

• New Technical Approach to represent OO Frameworks using SW
  – Courteous Inheritance: default rules increases reuse in ontologies

• New Strategy: go where the knowledge already is, then work outwards
  – Begin with MIT Process Handbook – open-source version in development
    • Example: process knowledge about selling
  – Future: Transformational wrappers around various legacy OO frameworks

• Roadmapping Market Evolution
  – Early adopters, creators, catalysts
  – Strategic players, forces
Larger Approach: Transformation Wrappers for OO Frameworks

- New Strategy: go where the knowledge already is, then work outwards
- Future: Transformational wrappers around various legacy OO frameworks
  - C++
  - Java, C#
  - UML
- Can use XSLT, SW tools, and/or XQuery engines to implement the transformations, guided by SWS ontology standardization practices
Outline of Talk

• Intro: Research on Semantic Web Services (SWS), its Business Uses
  – Rules, contracting, trust, policies
  – Integration, knowledge representation, standards

• Problem: Reusable Knowledge to Describe Services
  – Technique: knowledge representation to standardize on
  – Content investment: how to leverage legacy business process K

• New Technical Approach to represent OO Frameworks using SW
  – Courteous Inheritance: default rules increases reuse in ontologies

• New Strategy: go where the knowledge already is, then work outwards
  – Begin with MIT Process Handbook – open-source version in development
    • Example: process knowledge about selling
  – Future: Transformational wrappers around various legacy OO frameworks

• Roadmapping Market Evolution
  – Early adopters, creators, catalysts
  – Strategic players, forces
Some relevant example companies

- **Users:** *Amazon, Fidelity, Boeing; UPS, GM, Orbitz, eBay*

- **Vendors:** IBM, Microsoft, Oracle, HP, BEA, SAP; Sun, *Compiere*

- **Standards-oriented organizations:** *SWSI, BPMI, OPHI, UN CEFACCT*
Brief Tour of some More relevant websites

- [http://www.orbitz.com](http://www.orbitz.com) Orbitz, e.g., their vacation travel packages
- [http://www.compiere.org](http://www.compiere.org) Compiere open source ERP
Market Evolution: Discussion

Questions

• Existing and prospective early adopters

• Importance of open source content: seems to be an assumption/axiom for many people

• Prospective sources of open source content
Strategy Questions for Discussion

• ? Who/players: adopters, creators, catalysts?

• ? What forces/drivers for acceleration of adoption or investment, vs. inertia?

• ? Which additional interesting questions?
Yet More Discussion Questions:
Early Adoption Application
Prospects for SWS

• What business applications do you think are likely or interesting?
  – By vertical industry domain, e.g., health care or security
  – By task, e.g., authorization
  – By kind of shared information, e.g., patient records
  – By aspect of business relationships, e.g., provider network
• What do you think are entrepreneurial opportunity areas?
WRAP-UP: Outline of Talk

- Intro: Research on Semantic Web Services (SWS), its Business Uses
  - Rules, contracting, trust, policies
  - Integration, knowledge representation, standards
- Problem: Reusable Knowledge to Describe Services
  - Technique: knowledge representation to standardize on
  - Content investment: how to leverage legacy business process K
- New Technical Approach to represent OO Frameworks using SW
  - Courteous Inheritance: default rules increases reuse in ontologies
- New Strategy: go where the knowledge already is, then work outwards
  - Begin with MIT Process Handbook – open-source version in development
    - Example: process knowledge about selling
  - Future: Transformational wrappers around various legacy OO frameworks
- Roadmapping Market Evolution
  - Early adopters, creators, catalysts
  - Strategic players, forces

4/14/2004 Copyright 2002-2004 by Benjamin Grosof. All Rights Reserved.
OPTIONAL BACKUP SLIDES FOLLOW

• About early adopter prospects in SWS
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
SWS Adoption Roadmap: Strategy Considerations

• Expect see beginning in a lot of B2B interoperability or heterogeneous-info-integration intensive (e.g., finance, travel)
  – Actually, probably 1st intra-enterprise, e.g., EAI
• Reduce costs of communication in procurement, operations, customer service, supply chain ordering and logistics
  – increase speed, creates value, increases dynamism
  – macro effects create
    • stability sometimes (e.g., supply chain reactions due to lag; other negative feedbacks)
    • volatility sometimes (e.g., perhaps financial market swings)
  – increase flexibility, decrease lock-in
• Agility in business processes, supply chains
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)
END OF
OPTIONAL BACKUP SLIDES

• About early adopter prospects in SWS
• About Presenter’s SWS Research Agenda
Quickie Bio of Presenter

- MIT Sloan professor since 2000
- 12 years at IBM T.J. Watson Research; 2 years at startups
- PhD Comp Sci, Stanford; BA Applied Math Econ/Mgmt, Harvard
- Semantic web services is main research area:
  - Rules as core technology
  - Business Applications, Implications, Strategy:
    - e-contracting/supply-chain; finance; trust; …
  - Overall knowledge representation, e-commerce, intelligent agents
- Co-Founder, Rule Markup Language Initiative – the leading emerging standards body in semantic web rules (http://www.ruleml.org)
- Core participant in Semantic Web Services Initiative – which coordinates world-wide SWS research and early standards (http://www.swsi.org)
  - Area Editor for Contracts & Negotiation, Language Committee
  - Co-Chair, Industrial Partners program (SWSIP)
More about our SWS Technical Research Agenda

• Requirements Analysis (Biz → Tech)
  – New Application scenarios: e.g., SweetDeal e-contracting
  – Integrating rules, ontologies from many sources
  – Interoperability, power, consistency, scaleability

• New Fundamental Theory (Theory → Tech)
  – Description Logic Programs: bridging rules and ontologies
  – Situated Logic Programs: hooking rules to services
  – Courteous Logic Programs: prioritized conflict handling

• More:
  – Contributions to Early Standards Efforts: RuleML, SWSI
  – Piloting Early Adopter Areas: E-Contracts/SCM, Finance, Travel
  – Strategy Considerations and Implications
Analysis:
High-Level Requirements for SWS

• Support Biz-Process Communication
  – E.g., B2B SCM, CRM
  – E.g., e-contracts, financial info, trust management.

• Support SWS Tasks above current WS layers:
  – Discovery/search, invocation, deal negotiation, selection, composition, execution, monitoring, verification
New Analysis:
Key Technical Requirements for SWS

1. Combine rules with ontologies, from many web sources, with:
   - Rules on top of ontologies
   - Interoperability of heterogeneous rule and ontology systems
   - Power in inferencing
   - Consistency wrt inferencing
   - Scaleability of inferencing

2. Hook rules (with ontologies) up to web services
   - Ex. web services: enterprise applications, databases
   - Rules use services, e.g., to query, message, act with side-effects
   - Rules constitute services executably, e.g., workflow-y business processes
   - Rules describe services non-executably, e.g., for discovery, deal negotiation
   - On top of web service process models, coherently despite evolving messiness
3 Areas of New Fundamental KR Theory that enable Key Technical Requirements for SWS

• 1. Description Logic Programs:
  KR to combine LP (RuleML) rules on top of DL (OWL) ontologies, with:
  – Power in inferencing (including for consistency)
  – Scaleability of inferencing

• 2. Situated Logic Programs:
  KR to hook rules (with ontologies) up to (web) services
  – Rules use services, e.g., to query, message, act with side-effects
  – Rules constitute services executably, e.g., workflow-y business processes

• 3. Courteous Logic Programs:
  KR to combine rules from many sources, with:
  – Prioritized conflict handling to enable consistency, modularity; scaleably
  – Interoperable syntax and semantics
• About Semantic Web, Web Services
Some Semantic Web Advantages for Biz

- Builds upon XML’s much greater capabilities (vs. HTML*) for structured detailed descriptions that can be processed automatically.
  - Eases application development effort for assimilation of data in inter-enterprise interchange
- Knowledge-Based E-Markets -- where Agents Communicate (Agent = knowledge-based application)
  - ∴ potential to revolutionize interactivity in Web marketplaces: B2B, …
- Reuse same knowledge for multiple purposes/tasks/app’s
  - Exploit declarative KR; Schemas

* new version of HTML itself is now just a special case of XML

4/14/2004 Copyright 2002-2004 by Benjamin Grosof. All Rights Reserved.
**SWS Language effort**, on top of Current WS Standards Stack

**“Wire” Protocols**

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

**Service Description**

- **SOAP Blocks**
- **SOAP/XMLP**
- **XML**
- **HTTP/SMTP**
- **TCP/IP**

**SWS Language**

- Process
- WSDL Extensions
- WSDL
- XML

**Registry (UDDI)**

**Inspection**

**SWS Initiative (SWSI)** -- automate Tasks of:

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

[Slide authors: Benjamin Grosof (MIT Sloan), Sheila McIlraith (Stanford), David Martin (SRI International), James Snell (IBM)]

4/14/2004 Copyright 2002-2004 by Benjamin Grosof. All Rights Reserved.