Rule-based Semantic Services: Leveraging Knowledge Representation to Automate Business Processes

Prof. Benjamin Grosof

MIT Sloan School of Management

Information Technologies group

http://ebusiness.mit.edu/bgrosof

Invited Presentation (1-hour) at IBM Almaden Research Center,

San Jose, CA, USA, Jan. 30, 2006

Hosted by Savitha Srinivasan and James Spohrer

of Almaden Services Research (ASR)

Quickie Bio of Presenter Benjamin Grosof

- MIT Sloan professor since 2000
- 12 years at IBM T.J. Watson Research; 2 years at startups
- PhD Comp Sci (AI), Stanford; BA Applied Math Econ, Harvard
- Leader for 3 major software projects: 1 open source, 2 commercial
- Semantic+web+services is main research area:
  - Rules as core technology
  - Business Applications, Implications, Strategy:
    - e-contracting incl. ad,shop,monitor; trust; finance; …
  - Overall knowledge representation, e-commerce, intelligent agents
    - Learning. Hybrid techniques with onto, prob, induction.
- Co-Founder, Rule Markup Language Initiative – the leading emerging standards-design in semantic web rules (http://www.ruleml.org)
- Co-Editor, Semantic Web Services Initiative – which coordinates world-wide SWS research and early standards design (http://www.swsi.org)

Resources

- This slideset
  - Some of it will only be skimmed in this presentation
- Author's website (http://ebusiness.mit.edu/bgrosof)

  … see especially there:
  - Recent talks (including this one soon), not just papers
  - ISWC-2005 Tutorial slideset
  - SweetRules toolset (http://sweetrules.projects.semwebcentral.org)

Outline of Session

- Intro: What is the Semantic Web
  - Knowledge Representation in XML; Agents; with Web Services
- Why it Matters for Business
  - Knowledge-based Services Engineering
  - Examples of Policies for Contracting and Authorization
    - Pricing, Comparison Shopping, Ordering Lead Time, …
- Semantic Rules: Technology and Standardization
  - RuleML, Theory Advances, SweetRules Open Source Platform
- Roadmapping Business Value and Market Evolution
  - Cheaper, Faster, Better; EAI and B2B; Early Verticals
- Future Directions & Opportunities pertinent to ASR/SSME

The Semantic Web

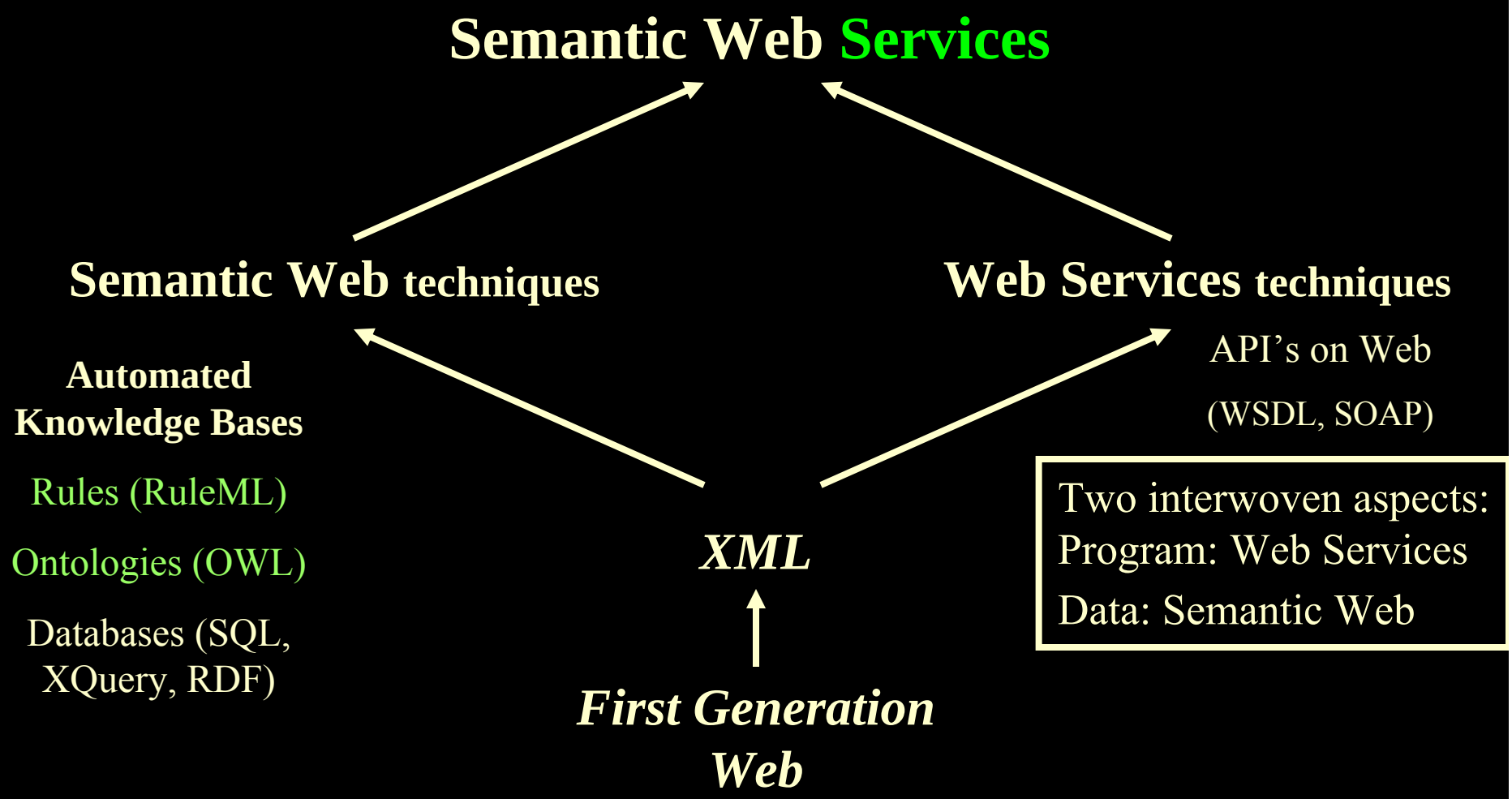
- The 1st generation, the Internet, enabled disparate machines to exchange data.
- The 2nd generation, the World Wide Web, enabled new applications on top of the growing Internet, making enormous amounts of information available, in human-readable form, and allowing a revolution in new applications, 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.

  - "Agent" = knowledge-based application

Next Generation Web



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

Vision of Evolution: Agents in Knowledge-Based E-Markets

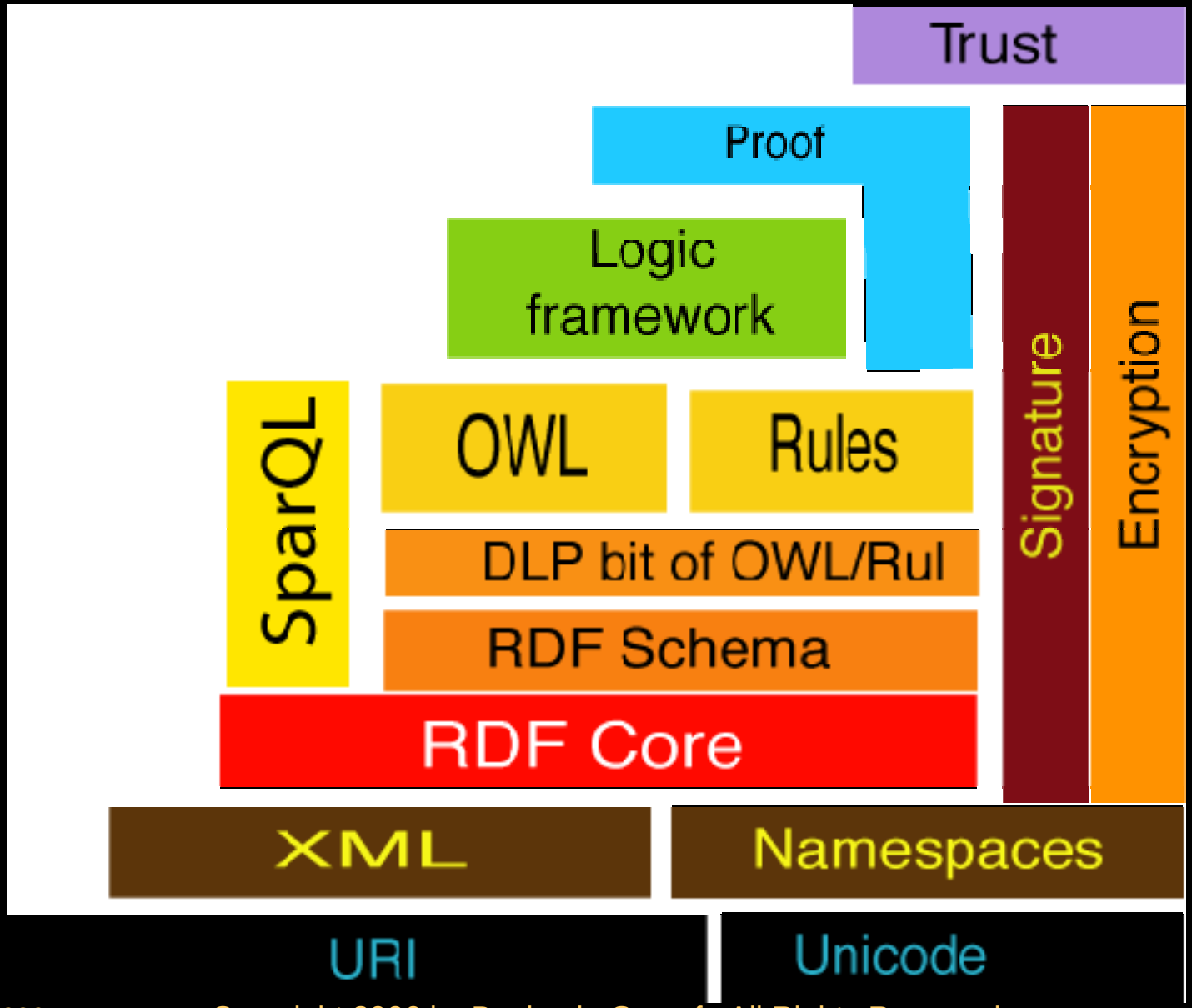
Coming soon to a world near you:...

- 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*

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 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 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
    - Also provide FOL KR

2005 W3C Semantic Web "Stack": Standardization Steps



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

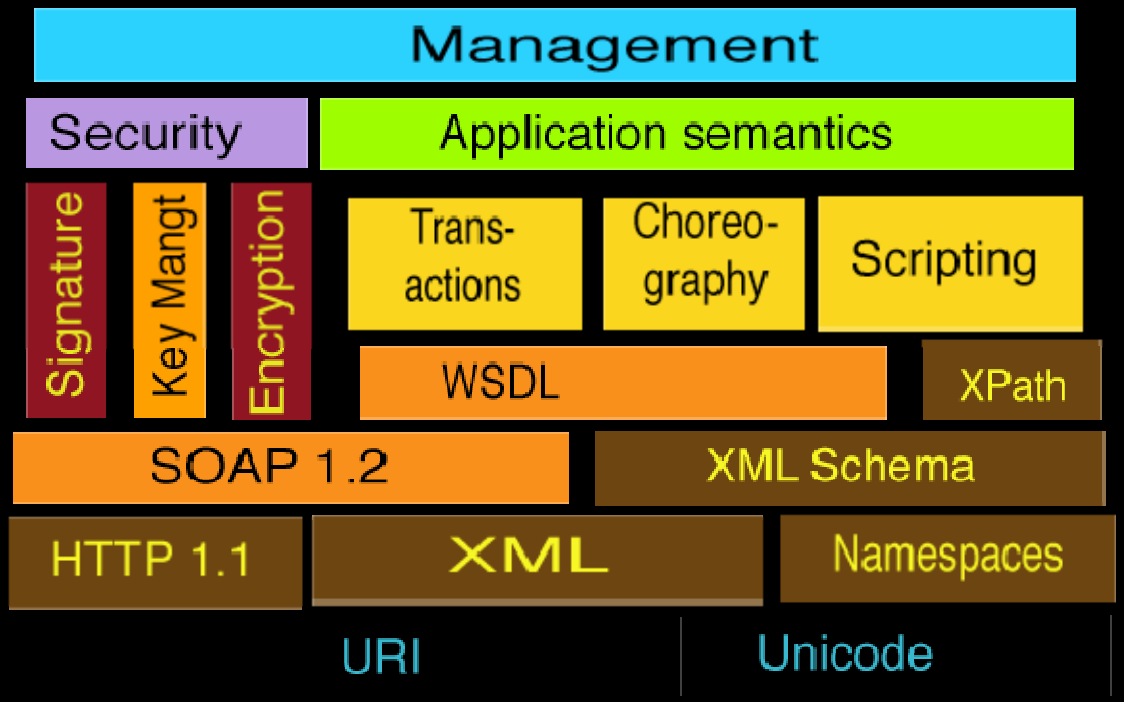
SW Overall Dependencies

- The W3C "stack" picture is a rough simplification.
- Rules do not require RDF
  - Can just use XML or even an ASCII "presentation syntax"
- Ontologies do not require RDF nor OWL
  - There are other techniques; OWL lacks some features
    - OWL does require RDF
- Customers and major vendors will be still digesting XML data management in next 2-5 years
  - … before moving on to heavy RDF usage

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
    - Advantage: **reuse** of knowledge across app's, these tasks
  - Integrated knowledge
- (Semantic Web) Services: e.g., infrastructural
  - Knowledge/info/DB integration
  - Inferencing and translation

Web Services Stack outline

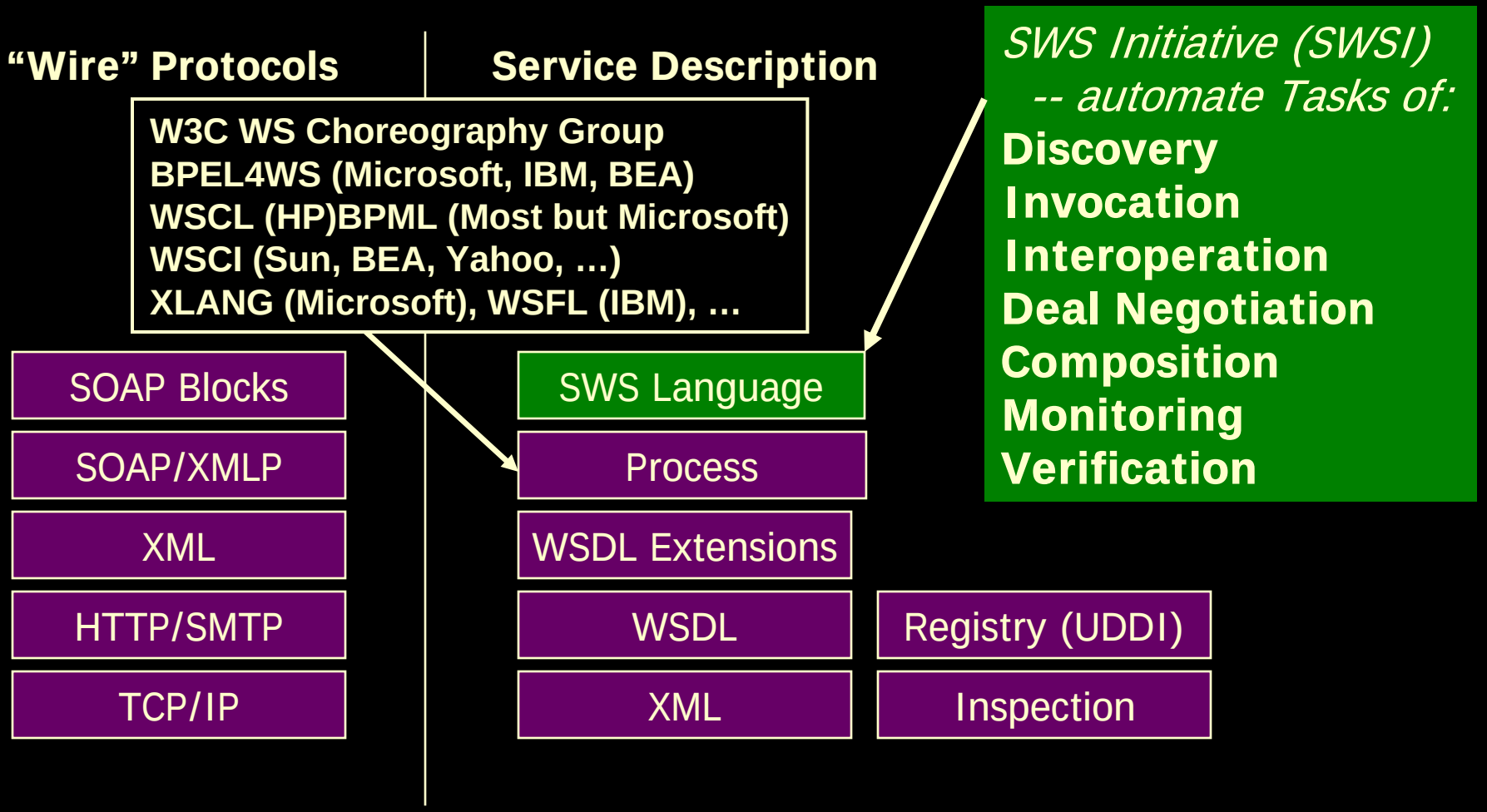


NOTES:

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

SWS Language effort, on top of Current WS Standards Stack



[Slide authors: Benjamin Grosof (MIT Sloan), Sheila McIlraith (Stanford) , David Martin (SRI International), 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
  - There's now a W3C Working Group
- "Agreement" here = agreement between invoker and provider of the service, described at knowledge level
- *Overall: in 2001-2002 lots of proprietary jockeying and de-facto mode testing/pressuring of the open-consortial standards bodies (e.g., of W3C) "riding the tiger". Then more via W3C, Oasis starting in 2003.*

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)]

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

Concept of Knowledge Representation (KR)

- A knowledge representation S is defined as a triple (LP, LC, |=), where:
  - LP is a formal language of sets of premises (i.e., premise expressions)
  - LC is a formal language of sets of conclusions (i.e., conclusion expressions)
  - |= is the entailment relation.
    - Conc(P,S) stands for the set of conclusions that are entailed in KR S by a set of premises P
    - We assume here that |= is a functional relation.

Example of Entailment: Mortality

- In First-Order Logic (FOL) KR:
  - Let P be the premises:
  - ∀?X. human(?X) ⇒ mortal(?X).
  - human(Socrates).
  -
  - In FOL, P entails (among others) the conclusion:
    - mortal(Socrates).
  - Notation:
    - "∀" means "for all".
    - "?" Prefixes a logical variable.

Example of Entailment: Sunday Stroll

- In Bayesian Probability KR:
  - Let P be the premises:
    - prob(rainySunday) = 0.4.
    - prob(funSunday | rainySunday) = 0.3.
    - prob(funSunday | ¬rainySunday) = 0.9.
  -
  - In this KR, P entails (among others) the conclusion:
    - prob(funSunday) = 0.66.

Example of Entailment: Discounting

- In the Courteous Logic Programs KR (e.g., RuleML):
  Let P be the premises:
  - {loyald} discount(?cust, RamadaHotel, 10percent)
    ← memberOf(?cust, AAA).
  - {seniord} discount(?cust, RamadaHotel, 25percent)
    ← age(?cust, ?x) and greaterThan(?x, 64).
  - overrides(seniord, loyald).
  - $\perp \leftarrow$ discount(?c, ?h, ?y) and discount(?c, ?h, ?z) | (?y $\neq$?z).
  - memberOf(Faisal, AAA).
  - age(Faisal, 72).

  - In this KR, P entails (among others) the conclusion:
    discount(Faisal, RamadaHotel, 25percent).

Example of Discounting, cont.'d

In the more general Production Logic Programs KR:

Suppose one adds the rule:

– @emailCouponAd(?cust, RamadaHotel, ?x)
← discount(?cust, RamadaHotel, ?x).

Then P entails the action (i.e., sanctions a call to an attached procedure):

@emailCouponAd(Faisal, RamadaHotel, 25percent).

KR: What's the Game? Desiderata

- Expressiveness: what can be said
  - useful, natural, complex enough
- Syntax: encoding data format -- e.g., in XML
  - easy enough to edit and communicate, by computers and by humans
- Semantics: principles of sanctioned inference, independent of reasoning algorithms:
  - clear, useful, natural, and understandable enough
- Computational Tractability (esp. worst-case): scale up in a manner qualitatively similar to relational databases: computation cycles go up as a polynomial function of input size
- Reasoning algorithms (compute the entailed conclusions):
  - sound (correct), complete, efficient, clear, and simple enough to engineer

Our Research Aspects/Questions about the Semantic Web

- Core technologies: Requirements, concepts, theory, algorithms, standards?
  - Rules in combination with ontologies; probabilistic, decision-/game-theoretic

- Business applications and implications: concepts, requirements analysis, techniques, scenarios, prototypes; strategies, business models, market-level evolution?
  - End-to-end e-contracting, finance, trust; …

Outline of Session

- Intro: What is the Semantic Web
  - Knowledge Representation in XML; Agents; with Web Services
- Why it Matters for Business
  - Knowledge-based Services Engineering
  - Examples of Policies for Contracting and Authorization
    - Pricing, Comparison Shopping, Ordering Lead Time, …
- Semantic Rules: Technology and Standardization
  - RuleML, Theory Advances, SweetRules Open Source Platform
- Roadmapping Business Value and Market Evolution
  - Cheaper, Faster, Better; EAI and B2B; Early Verticals
- Future Directions & Opportunities pertinent to ASR/SSME

Some Answers to: "Why does SW Matter to Business?"

- 1. "Death. Taxes. Integration." - They're always with us.
- 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

Strategic Business Foci in our SW Research

- Knowledge-based Services Engineering: intra- and inter- enterprise
- Target "killer app" known for 30 years: do better job of EDI
- Challenges:
  - Ease of development, deployment ↑
  - Reuse of knowledge ↑
  - ⇒ life cycle costs ↓ , agility ↑
- Starting with: Policies
  - Using recent theory breakthroughs in semantic rules
  - E.g., for end-to-end contracting and authorization (incl. security)
- Starting with: EAI as well as B2B

SW Rules: Use Cases from our research

- Contracts/negotiation, advertising/discovery
  - E-procurement, E-selling
  - Pricing, terms & conditions, supplier qualification, …
- Monitoring:
  - Exception handling, e.g., of contract violations
    - Late delivery, refunds, cancellation, notifications
  - Notifications, personal messaging, and other workflow
- Trust Policies: authorization, confidentiality & privacy, security, access control
  - E.g., financial services, health care
    - *Extensive analysis of business case/value*
- Semantic mediation: rule-based ontology translation, context-based information integration

SWS and Rules Summary

*** SWS Tasks Form 2 Distinct Clusters,*

each with associated Central Kind of Service-description Knowledge and Main KR

1. Security/Trust, Monitoring, Contracts, Advertising/Discovery, Ontology-mapping Mediation
   - Central Kind of Knowledge: Policies
   - Main KR: Nonmon LP (rules + ontologies)
2. Composition, Verification, Enactment
   - Central Kind of Knowledge: Process Models
   - Main KR: FOL (axioms + ontologies)
     - + Nonmon LP for ramifications (e.g., cf. Golog)

- Thus RuleML & SWSF specify both Rules, FOL
  - Fundamental KR Challenge: "Bridging" Nonmon LP with FOL
    - SWSF experimental approach based on hypermon. [Grosof & Martin]

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

EECOMS Example of SCM Policy Rules: Ordering Lead Time

- Vendor's rules that prescribe how buyer must place or modify an order:
- A) 14 days ahead if the buyer is a qualified customer.
- B) 30 days ahead if the ordered item is a minor part.
- C) 2 days ahead if the ordered item's item-type is backlogged at the vendor, the order is a modification to reduce the quantity of the item, and the buyer is a qualified customer.

- Suppose more than one of the above applies to the current order? **Conflict!**

- Helpful Approach: **precedence** between the rules. Often only *partial* order of precedence is justified. E.g., C > A.

Courteous LP's: Ordering Lead Time Example

{leadTimeRule1} orderModificationNotice(?Order,14days)
← preferredCustomerOf(?Buyer,?Seller) ∧
purchaseOrder(?Order,?Buyer,?Seller) .

{leadTimeRule2} orderModificationNotice(?Order,30days)
← minorPart(?Buyer,?Seller,?Order) ∧
purchaseOrder(?Order,?Buyer,?Seller) .

{leadTimeRule3} orderModificationNotice(?Order,2days)
← preferredCustomerOf(?Buyer,?Seller) ∧
orderModificationType(?Order,reduce) ∧
orderItemIsInBacklog(?Order) ∧
purchaseOrder(?Order,?Buyer,?Seller) .

overrides(leadTimeRule3 , leadTimeRule1) .

⊥ ← orderModificationNotice(?Order,?X) ∧
orderModificationNotice(?Order,?Y) | (?X ≠?Y) .

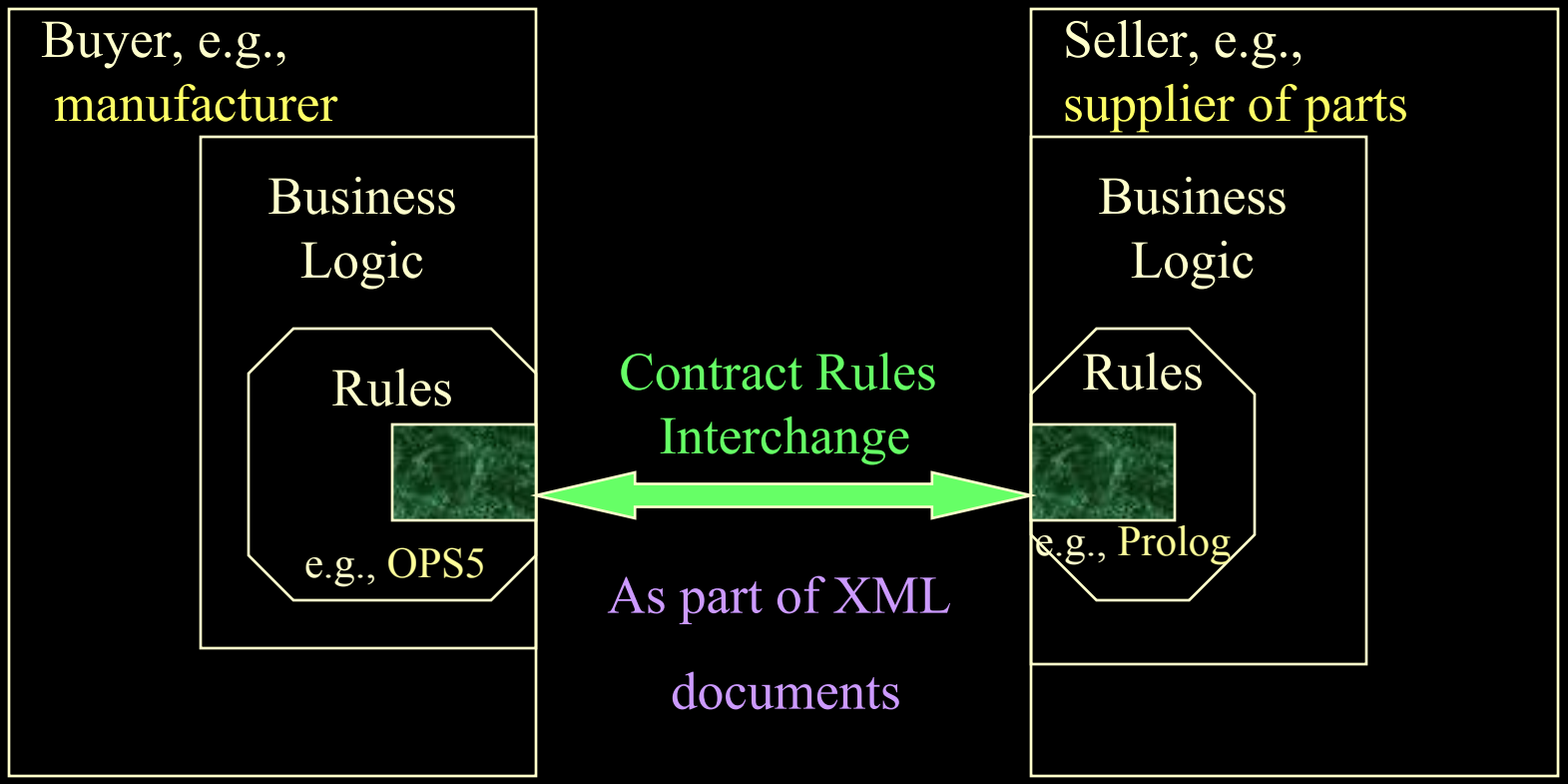
End-to-End E-Contracting Tasks

- Discovery, advertising, matchmaking
  - Search, sourcing, qualification/credit checking
- Negotiation, bargaining, auctions, selection, forming agreements, committing
  - Hypothetical reasoning, what-if'ing, valuation
- Performance/execution of agreement
  - Delivery, payment, shipping, receiving, notification
- Problem Resolution, Monitoring
  - Exception handling

SweetDeal Approach: Rule-based Contracts for E-commerce

- Rules as way to specify (part of) business processes, policies, products: as (part of) contract terms.
- Complete or partial contract.
  - As default rules. Update, e.g., in negotiation.
- Rules provide high level of conceptual abstraction.
  - easier for non-programmers to understand, specify, dynamically modify & merge. E.g.,
  - by multiple authors, cross-enterprise, cross-application.
- Executable. Integrate with other rule-based business processes.

Contract Rules during Negotiation

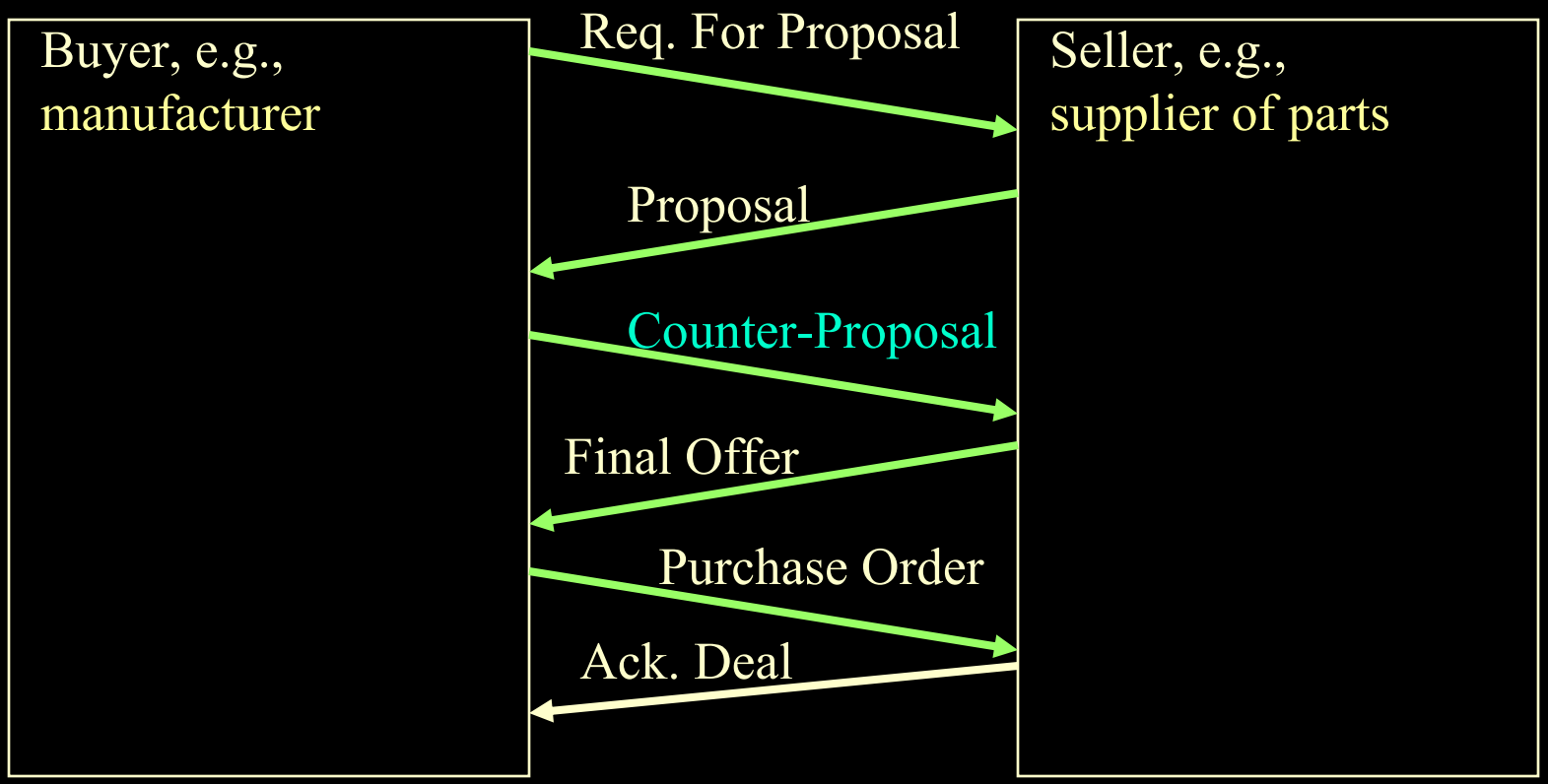


Contracting parties NEGOTIATE via shared rules.

Examples of Contract Provisions Well-Represented by Rules in Automated Deal Making

- Product descriptions
  - Product catalogs: properties, conditional on other properties.
- Pricing dependent upon: delivery-date, quantity, group memberships, umbrella contract provisions
- Terms & conditions: refund/cancellation timelines/deposits, lateness/quality penalties, ordering lead time, shipping, creditworthiness, biz-partner qualification, service provisions
- Trust
  - Creditworthiness, authorization, required signatures
- *Buyer Requirements (RFQ, RFP) wrt the above*
- *Seller Capabilities (Sourcing, Qualification) wrt the above*

Exchange of Rules Content during Negotiation: example



Example: E-Contract Proposal from supplierCo to manufCo

- …
  {usualPrice} price(per_unit, ?PO, $60) ←
- purchaseOrder(?PO, supplierCo, ?AnyBuyer) ∧
- quantity_ordered(?PO, ?Q) ∧ (?Q ≥ 5) ∧ (?Q ≤ 1000) ∧
- shipping_date(?PO, ?D) ∧ (?D ≥ 24Apr00) ∧ (?D ≤ 12May00).
- {volumeDiscount} price(per_unit, ?PO, $51) ←
- purchaseOrder(?PO, supplierCo, ?AnyBuyer) ∧
- quantity_ordered(?PO, ?Q) ∧ (?Q ≥ 100) ∧ (?Q ≤ 1000) ∧
- shipping_date(?PO, ?D) ∧ (?D ≥ 28Apr00) ∧ (?D ≤ 12May00) .
  overrides(volumeDiscount , usualPrice) .
- ⊥ ← price(per_unit, ?PO, ?X) ∧ price(per_unit, ?PO, ?Y) GIVEN (?X ≠ ?Y).
- ...

Negotiation Ex. Doc. Rules: Counter-Proposal *from manufCo to supplierCo*

- …
  {usualPrice} price(per_unit, ?PO, $60) ← ...
- {volumeDiscount} price(per_unit, ?PO, $51) ←
- purchaseOrder(?PO, supplierCo, ?AnyBuyer) ∧
- quantity_ordered(?PO, ?Q) ∧ (?Q ≥ 5) ∧ (?Q ≤ 1000) ∧
- shipping_date(?PO, ?D) ∧ (?D ≥ 28Apr00) ∧ (?D ≤ 12May00) .
  overrides(volumeDiscount , usualPrice) .
- ⊥ ← price(per_unit, ?PO, ?X) ∧ price(per_unit, ?PO, ?Y) GIVEN (?X ≠ ?Y).
- {aSpecialDeal} price(per_unit, ?PO, $48) ←
- purchaseOrder(?PO, supplierCo, manufCo) ∧
- quantity_ordered(?PO, ?Q) ∧ (?Q ≥ 400) ∧ (?Q ≤ 1000) ∧
- shipping_date(?PO, ?D) ∧ (?D ≥ 02May00) ∧ (?D ≤ 12May00) .
- overrides(aSpecialDeal, volumeDiscount) .
- overrides(aSpecialDeal , usualPrice) .
- ...

Simply *added* rules!

Negotiation Example --

XML Encoding of Rules in RuleML

```
<rulebase>
 <imp>
  <rlab>usualPrice</_rlab>
  <head>
   <atom>
    <opr><rel>price</rel></_opr>
    <ind>per_unit</ind>
    <var>PO</var>
    <ind>$60</ind>
   </atom>
  </head>
  <body>   …  (see next page)   </_body>
 </imp>
…
</rulebase>
```

SweetDeal V2 Demo Outline

- SweetDeal E-Contracting Application using SweetRules (supply chain)
  - SCLP RuleML that includes OWL ontologies
  - Contract proposals/final-agreements are SCLP RuleML rulebases that reference/include OWL ontologies
  - Humans edit & communicate, supported by automated agents
  - Proposal evaluation supported by inferencing
  - Agreed business process is executable via inferencing+action

What Can Be Done with the Rules in contracting, & negotiation, based on our SweetDeal approach to rule representation

- Communicate: with deep shared semantics
  - via RuleML, inter-operable with same sanctioned inferences
  - ⇔ heterogeneous rule/DB systems / rule-based applications ("agents")
- Execute contract provisions:
  - infer; ebiz actions; authorize; ...
- Modify easily: contingent provisions
  - default rules; modularity; exceptions, overriding
- Reason about the contract/proposal
  - hypotheticals, test, evaluate; tractably
  - *(also need "solo" decision making/support by each agent)*

Outline of Session

- Intro: What is the Semantic Web
  - Knowledge Representation in XML; Agents; with Web Services
- Why it Matters for Business
  - Knowledge-based Services Engineering
  - Examples of Policies for Contracting and Authorization
    - Pricing, Comparison Shopping, Ordering Lead Time, …
- Semantic Rules: Technology and Standardization
  - RuleML, Theory Advances, SweetRules Open Source Platform
- Roadmapping Business Value and Market Evolution
  - Cheaper, Faster, Better; EAI and B2B; Early Verticals
- Future Directions & Opportunities pertinent to ASR/SSME

XBRL®
eXtensible Business Reporting Language

Welcome to XBRL International

16 - 17 NOVEMBER 2004 | 10th International
BRUSSELS | CONFERENC

Financial Reporting Goes Global - XBRL and IFRS Working Together

For more information, please visit the Conference Website register today.

XBRL is a language for the electronic communication of business financial data which is set to revolutionise business reporting arou the world. It provides major benefits in the preparation, analysis communication of business information. It offers cost savings, gre efficiency and improved accuracy and reliability to all those involv in supplying or using financial data.

XBRL stands for eXtensible Business Reporting Language. It is or a family of "XML" languages which is becoming a standard means communicating information between businesses and on the intern

XBRL is being developed by an international non-profit consortiun approximately 250 major companies, organisations and governm agencies. **It is an open standard, free of licence fees.** It is already being put to practical use in a number of countries and implementations of XBRL are growing rapidly around the world.

This site provides information about the nature, uses and benefits XBRL. It explains how individuals and companies can join the effo move forward and make use of the language.

Equational Ontological Conflicts

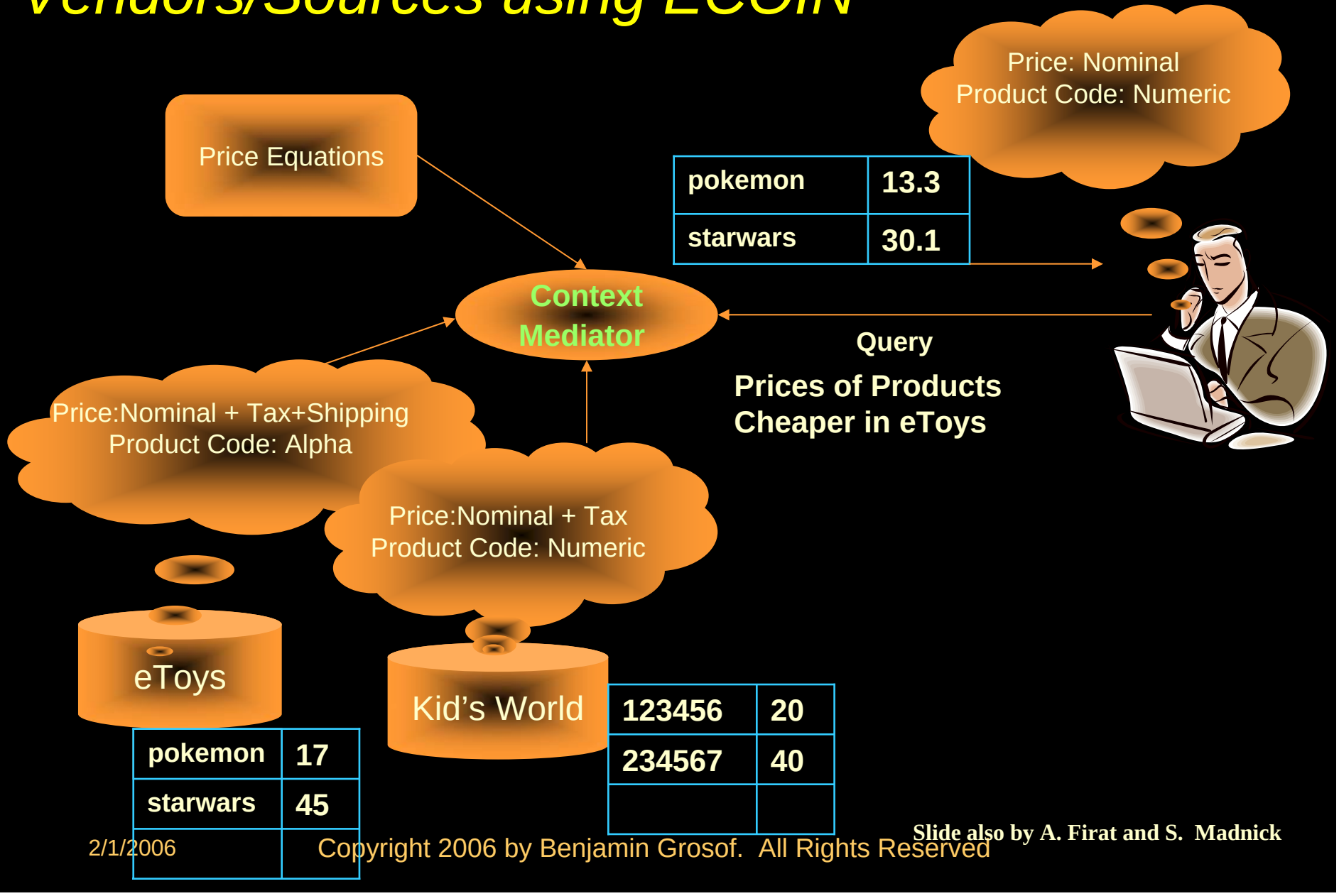
Key Concepts

Gross Profit = Net Sales – Cost of Goods	Gross Profit = Net Sales – Cost of Goods – Depreciation
Price = Nominal Price + Shipping	Price = Nominal Price + Shipping + Tax

" heterogeneity in the way data items are *calculated* from other data items *in terms of definitional equations*"

Slide also by A. Firat and S. Madnick

Comparing Prices From Multiple Vendors/Sources using ECOIN



Slide also by A. Firat and S. Madnick

Approach: ECOIN

- Extended COntext INterchange, developed at MIT Sloan
  - *[Firat, Madnick, & Grosof] (Best Paper Award WITS-2002)*
- Context-based loosely-coupled integration
  - *Extends the Context Interchange (COIN) framework also developed at MIT*
- Symbolic Equation Solving using Constraint Logic Programming
  - *Integrates symbolic equation solving techniques with abductive logic programming*
- *In-progress:* Utilizing RuleML and OWL in ECOIN

Slide also by A. Firat and S. Madnick

Outline of Session

- Intro: What is the Semantic Web
  - Knowledge Representation in XML; Agents; with Web Services
- Why it Matters for Business
  - Knowledge-based Services Engineering
  - Examples of Policies for Contracting and Authorization
    - Pricing, Comparison Shopping, Ordering Lead Time, …
- Semantic Rules: Technology and Standardization
  - RuleML, Theory Advances, SweetRules Open Source Platform
- Roadmapping Business Value and Market Evolution
  - Cheaper, Faster, Better; EAI and B2B; Early Verticals
- Future Directions & Opportunities pertinent to ASR/SSME

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.
- Known advantages of rules vs. general code
  - separable business logic, more reusable across app.'s, life cycle
  - good for loose coupling cf. workflow
  - good for representing contingent behavior of services/processes.
  - 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.

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.)*

SWS/Contracting Criteria for Rule Representation

1

2

3

- *High-level:* Agents reach common understanding; contract is easily modifiable, communicatable, executable.
- Inter-operate: heterogeneous commercially important rule systems.
- Expressive power, convenience, natural-ness.
- ... but: computational tractability.
- Modularity and locality in revision.
- Declarative semantics.
- Logical non-monotonicity: default rules, negation-as-failure.
  - essential feature in commercially important rule systems.
- Prioritized conflict handling.
- Ease of parsing.
- Integration into Web-world software engineering.
- Procedural attachments.

OLP

Courteous

XML

Situated

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

New Fundamental Rule KR Theory I
that enables Key Technical Requirements for SWS

- 1. Courteous Logic Programs: [Grosof]

  KR to combine rules from many sources, with:
  - Prioritized conflict handling to enable consistency, modularity; scaleably
  - Interoperable syntax and semantics

- 2. Situated Logic Programs: [Grosof]

  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

New Fundamental Rule KR Theory II
that enables Key Technical Requirements for SWS

- 3. Reference Ontologies from Rules Via URI Names [Grosof]

- 4. Description Logic Programs: [Grosof, Horrocks, Volz, & Decker]

  KR to combine LP (RuleML) rules on top of DL (OWL) ontologies,

  with:
  - Power in inferencing (including for consistency)
  - Scaleability of inferencing
  - Approach: Analyze and exploit the Intersection of DL and LP (within FOL)

New Fundamental Rule KR Theory III
that enables Key Technical Requirements for SWS

- 5. Courteous Inheritance: [Grosof & Bernstein]
  - OO default inheritance as Courteous LP
  - Used to Leverage Process Handbook, & other Legacy OO Knowledge, to create SW service ontologies
- 6. Production Rules as LP: [Grosof]
  - OPS5-heritage production rules as Situated Courteous LP
  - Find and fix fundamental weakness in chaining through negation in Rete-based inferencing
  - Unify commercially most important and fast-growing rule families
- 7. Hypermonotonic Reasoning: [Grosof (in-progress)]
  - Unify Nonmon LP KR with FOL KR
  - Nonmon LP as sound & incomplete wrt FOL

Slide also by Harold Boley (NRC)

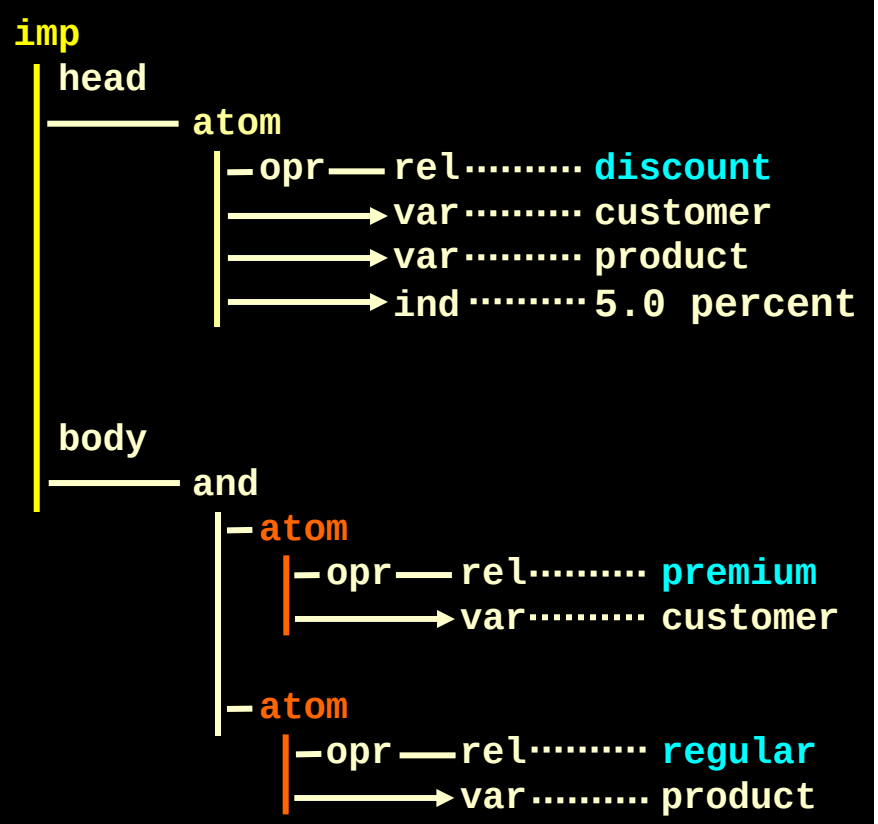
RuleML Example: Markup and Tree

'The **discount** for a *customer* buying a *product* is **5.0 percent** if the *customer* is **premium** and the *product* is **regular**.',

discount(?customer,?product,"5.0 percent") ← premium(?customer) /\ regular(?product);

Markup Syntax

```
<imp>
  <_head>
    <atom>
     <opr><rel>discount</rel></opr>
     <tup><var>customer</var>
          <var>product</var>
          <ind>5.0 percent</ind></tup>
    </atom>
  </head>
  <body>
    <and>
      <atom>
        <_opr><rel>premium</rel></opr>
        <tup><var>customer</var></tup>
      </atom>
      <atom>
        <_opr><rel>regular</rel></opr>
        <tup><var>product</var></tup>
      </atom>
    </and>
  </body>
</imp>
```



`tup` is an ordered tuple.

Semantic Rules News

News recently:

- Fundamental theory and technique breakthroughs, e.g.:
  - Declarative logic programs (LP) basis for interoperability, then webized → RuleML standards design (2001-)
  - Courteous LP prioritized defaults, robust modular merging
  - Description LP ontology integration
  - Production LP interoperability+semantics for production rules, declarative procedural attachments for actions and queries
  - SweetRules V2 open source toolset platform (2004-)
- Large US, EU research projects (DAML, WSMO) focus on rules (DARPA Agent Markup Language; Web Service Mediation Ontology)

Semantic Rules News (cont.'d)

News recently:

- W3C forms Rule Interchange Format WG, full standards effort, after holding a Workshop (2005)
- OMG forms standards efforts on production rules, rule management
- Semantic Web Services Framework design (2005) focuses on rules
- Rule-based Policy area heats up in web services, semantic web, incl. at Oasis. Oasis forms Semantic Execution Env. standards effort (2005).
- Semantic web rules workshop series becomes full research conference (RuleML-2005) colocated with ISWC
- Gartner etc. reports on rules sector

Semantic Rules: Differences from Rules in the 1980's / Expert Systems Era

- Get the KR right (knowledge representation)
  - More mature research understanding
  - Semantics independent of algorithm/implementation
  - Cleaner; avoid general programming/scripting language capabilities
  - Highly scaleable performance; better algorithms; choice from interoperability
  - Highly modular wrt updating; use prioritization
  - → Highly dynamic, scaleable rulebase authoring: distributed, integration, partnering
- Leverage Web, esp. XML
  - Interoperable syntax
  - Merge knowledge bases
- Embeddable
  - Into mainstream software development environments (Java, C++, C#); not its own programming language/system (cf. Prolog)
- Knowledge Sharing: intra- or inter- enterprise
- Broader set of Applications

Overview of RuleML Today I

- RuleML Initiative (2000--) http://www.ruleml.org
  - Dozens of institutions (~35), researchers; esp. in US+Canada, EU
- Mission priorities:
  1. Enable semantic exchange of rules/facts between most commercially important rule systems
     - Production rules, relational databases, Prolog, Event-Condition-Action rules
  2. Synergize with RDF, OWL (& other relevant web standards as arrive)
  3. Enable rule-based semantic web services, e.g., policies
- Standards specification: current version V0.8+
  - 1st version 2001; basic now fairly stable
- Logical Knowledge Representation at core of semantics
  - Declarative Logic Programs (LP) & First Order Logic (FOL) … Webized
  - Firm foundations in decades of R&D theory, algorithms, implementations

Overview of RuleML Today II

- A number of tools (~60 engines, translators, editors), demo applications. E.g., SweetRules open source platform.
- Very influential & lots of mindshare in cutting edge R&D community. 20,000+ Google Hits (as of Mar. 2004)
- Annual International Scientific Workshop since 2002
- Cooperating closely with the leading umbrella Web standards organizations and SW research efforts:
  - OMG – providing markup and semantics for production rules meta-model.
  - Discussions well underway to launch Oasis, W3C standards working groups.
  - Encouraged (and funded in part) by DARPA
  - Collaborating with Semantic Web Services Initiative (SWSL), Web Services Mediation Language (WSML) & REWERSE in EU

New Fundamental Rule KR Theory
that enables Key Technical Requirements for SWS

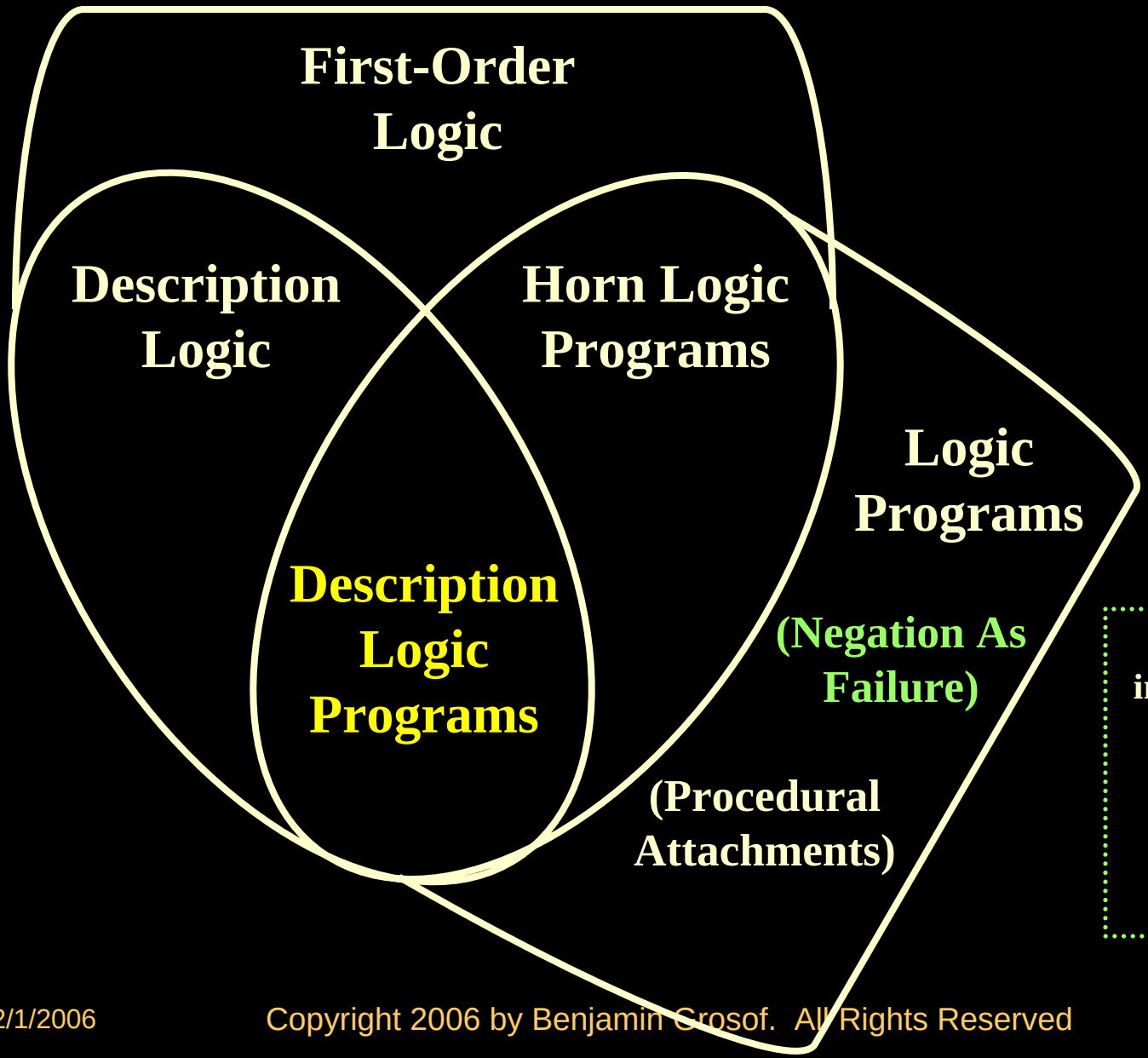
In 1985-94:

- Prolog interoperable with relational DB; LP extends core-SQL [many]
- Richer logical connectives, quantifiers [Lloyd & Topor]
- "Well Founded" Semantics for Negation-As-Failure [Van Gelder et al; Przmusinski]
- Hilog quasi-higher order expressiveness, meta-syntax flexibility [Kifer et al.]
- Frame syntax cf. F-Logic [Kifer *et al.*]

In 1995-2004:

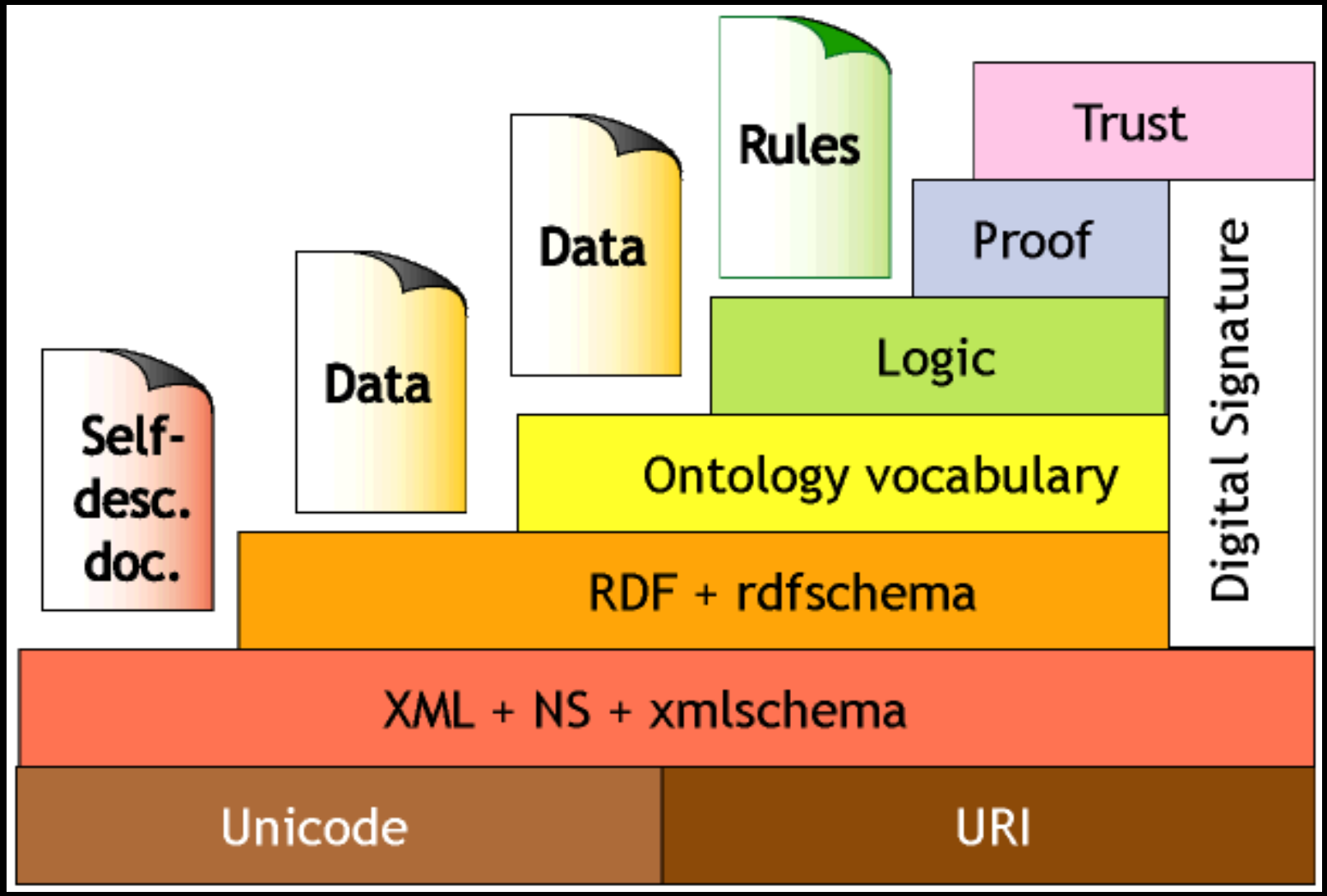
- Courteous LP: prioritized conflict handling [Grosof]
  - Robust, tractable, modular merging & updating
- Situated LP: hook rules up to services [Grosof]
- Description LP: combine Description Logic ontologies [Grosof *et al.*]
- Courteous Inheritance: combine OO default ontologies [Grosof *et al.*]
- Production Rules as LP: interoperate [Grosof *et al.*]
  - Declarative LP as interoperable core between commercial families [Grosof *et al.*]
- Hypermonotonic Reasoning: combine with FOL [Grosof (in-progress)]

Venn Diagram: Expressive Overlaps among KR's

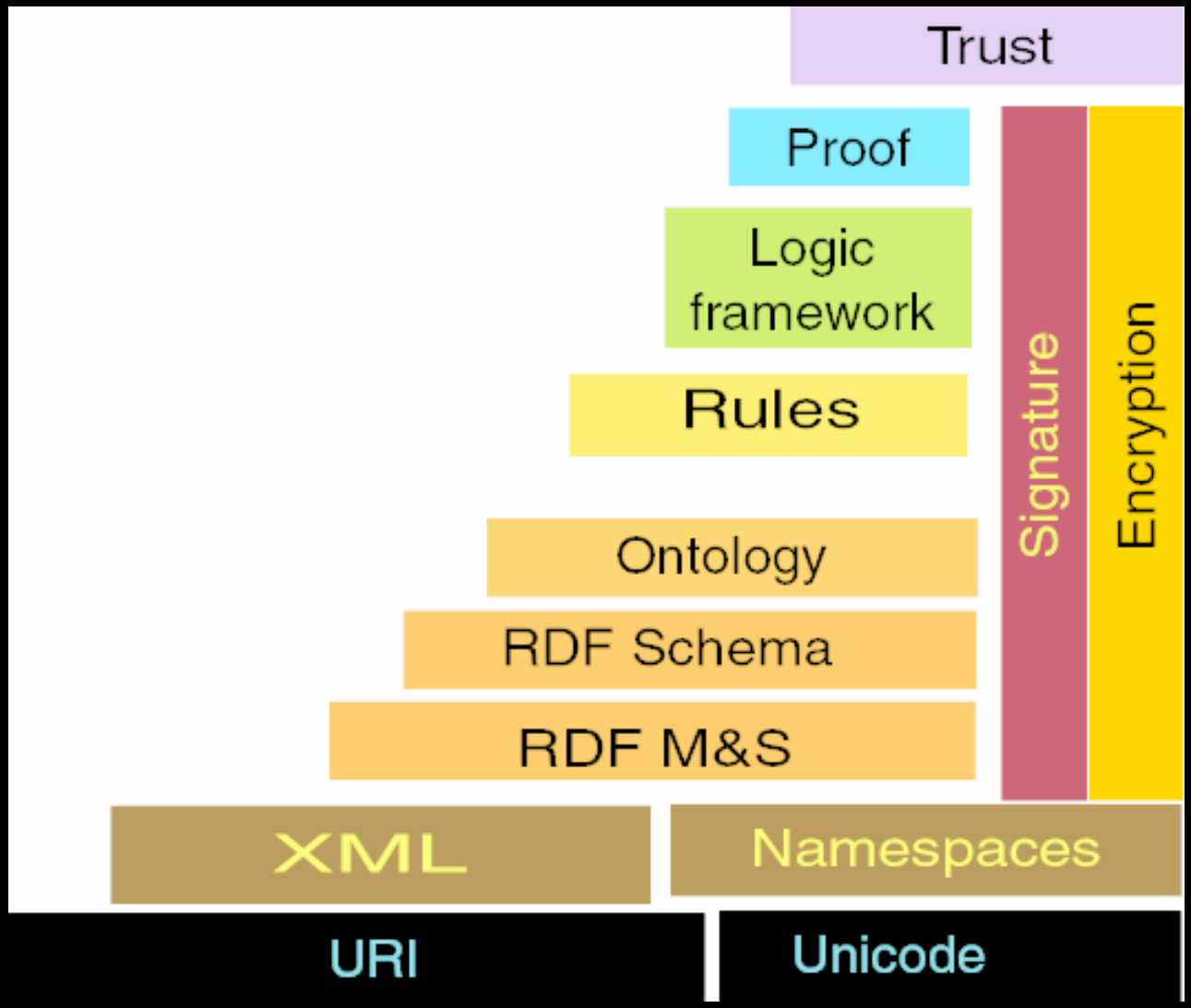


NB: Nonmon LP, including Courteous, relies on NAF as fundamental underlying KR expressive mechanism

2000 W3C Semantic Web "Stack": Standardization Steps

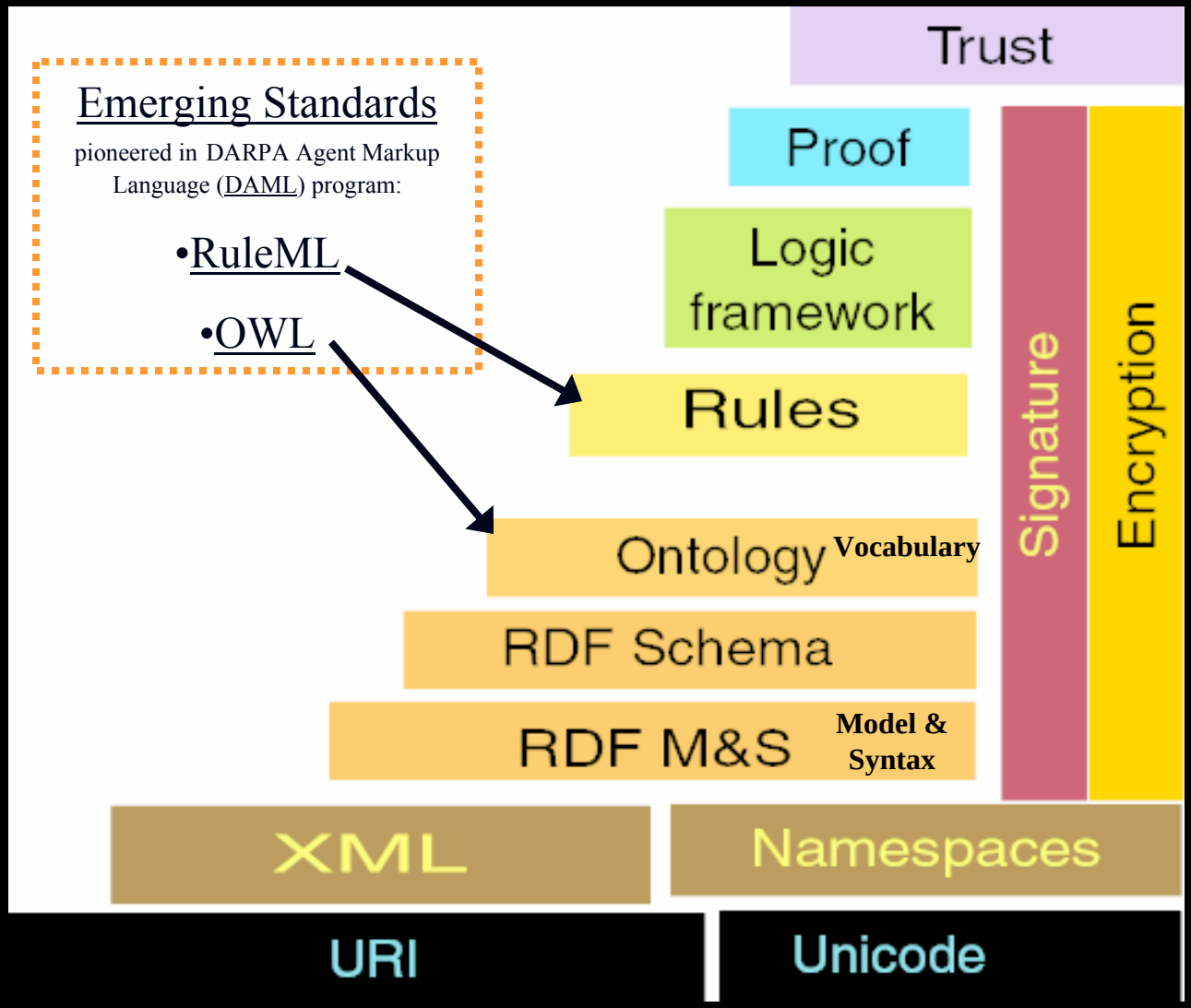


2002 W3C Semantic Web "Stack": Standardization Steps



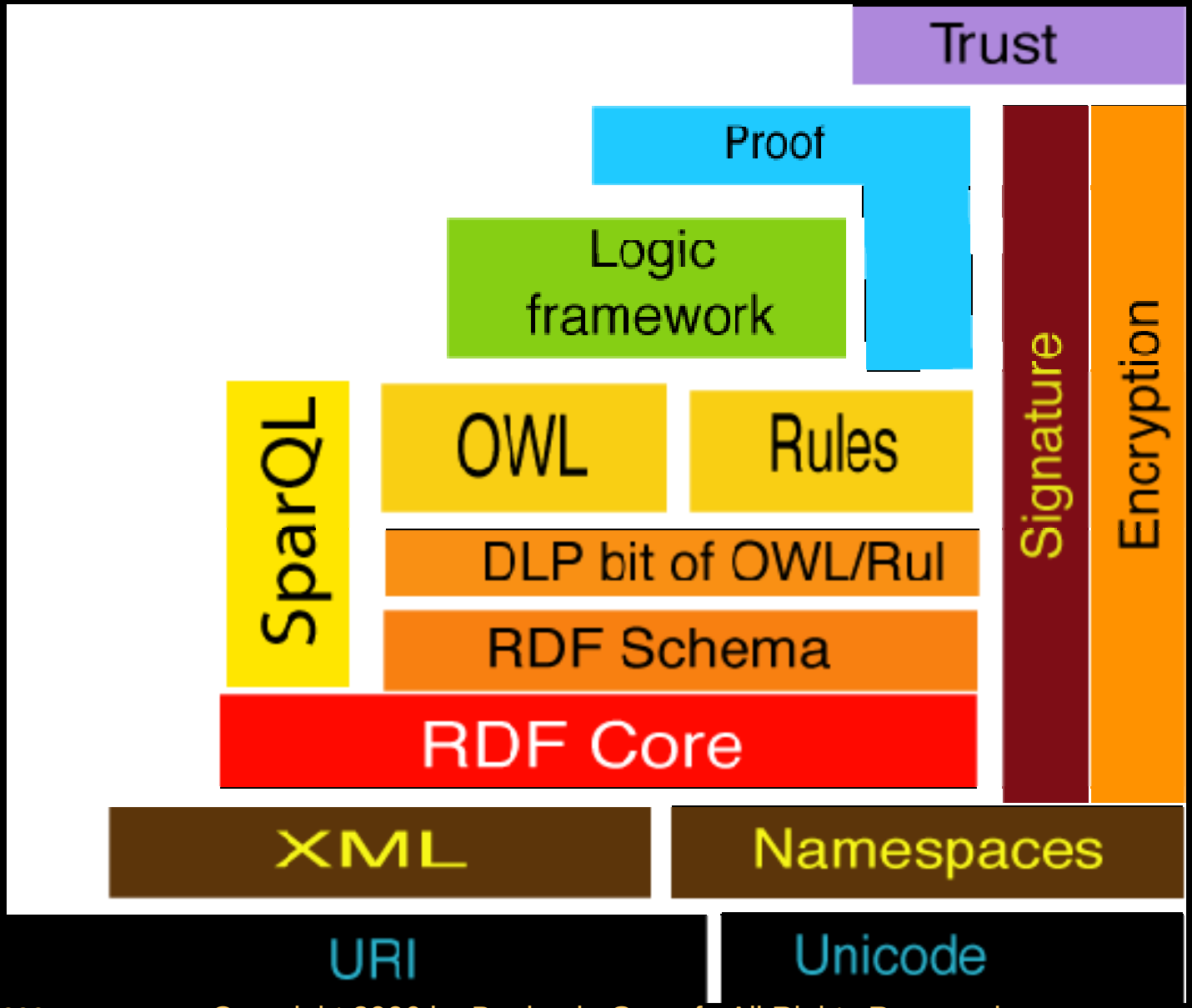
[Diagram http://www.w3.org/DesignIssues/diagrams/sw-stack-2002.png is courtesy Tim Berners-Lee]

2002 W3C Semantic Web "Stack": Standardization Steps



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

2005 W3C Semantic Web "Stack": Standardization Steps

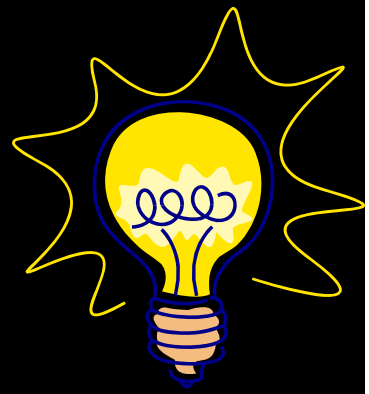


Changes in W3C SW Stack Diagram I

- 2000: Rules not viewed as part of Logic; is vaguely related
- 2002: Added Rules
  - as a semantic step/layer; and
  - as next after OWL ontologies in the architectural layering and standardization sequence
  - Based on RuleML and DAML rules research, and CommonRules tool
  - Underpinned by declarative LP as KR theory & approach
  - Based also on Notation 3 research and cwm tool

Changes in W3C SW Stack Diagram II

- 2005: Added Description Logic Programs (DLP) as common core architectural layer under both Rules and OWL.
  - Based on DLP research, SweetRules V2 tool, and W3C Workshop discussion consensus
  - *The only piece of research named in the diagram*



- DLP:
  - Invented in Aug. 2002 (1st working paper)
  - SweetRules V2 tool implemented it
  - Original use case & requirements came from SweetDeal e-contracting research incl. application scenario prototyping

Outline of Session

- Intro: What is the Semantic Web
  - Knowledge Representation in XML; Agents; with Web Services
- Why it Matters for Business
  - Knowledge-based Services Engineering
  - Examples of Policies for Contracting and Authorization
    - Pricing, Comparison Shopping, Ordering Lead Time, …
- Semantic Rules: Technology and Standardization
  - RuleML, Theory Advances, SweetRules Open Source Platform
- Roadmapping Business Value and Market Evolution
  - Cheaper, Faster, Better; EAI and B2B; Early Verticals
- Future Directions & Opportunities pertinent to ASR/SSME

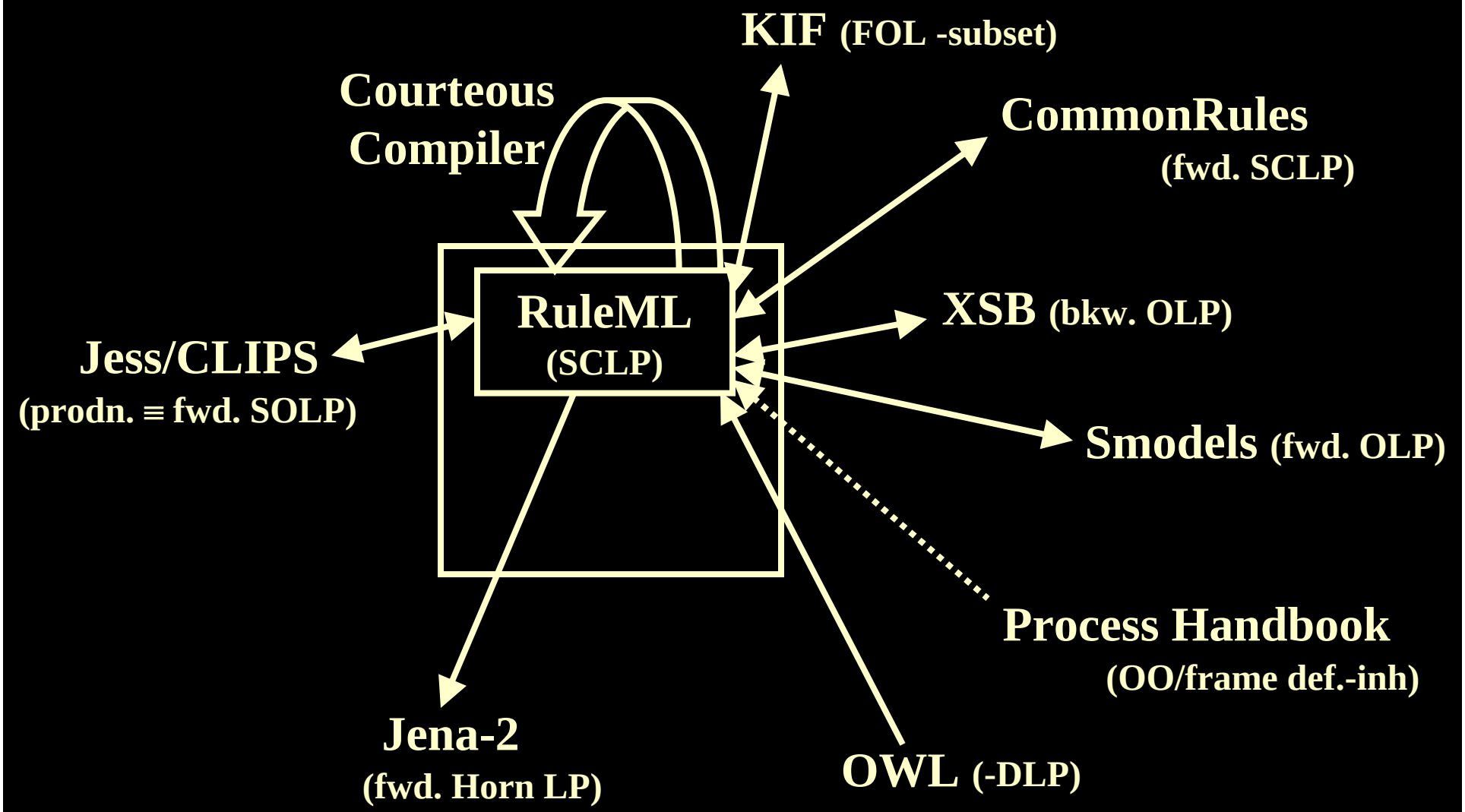
SweetRules Overview

- Concept and Architecture: Open Source Tools Platform for SW Rules and RuleML. http://sweetrules.projects.semwebcentral.org (2004-)
  - Multi-institutional collaboration led by MIT Sloan, with 12+ other co.'s / univ.'s
- Capabilities:
  - Translation and interoperability between heterogeneous rule systems (forward- and backward-chaining) and their rule languages/representations of the most commercially important flavors (relational database / Prolog and production rules / event-condition-action)
  - Inferencing including via translation between rule systems
  - Authoring, Analysis, and testing of rulebases
  - Open, lightweight, extensible, pluggable architecture overall
  - Merge knowledge bases
    - Combine rules with ontologies, incl. OWL, OO default inheritance
  - Focus on kinds of rule systems that are commercially important
    - E.g., Jess production rules, XSB Prolog, IBM Common Rules, HP Jena, …
  - Highly scaleable performance by piggybacking on mature commercial implementations (e.g., Jess, XSB)
  - Automatically composes translators, inference engines

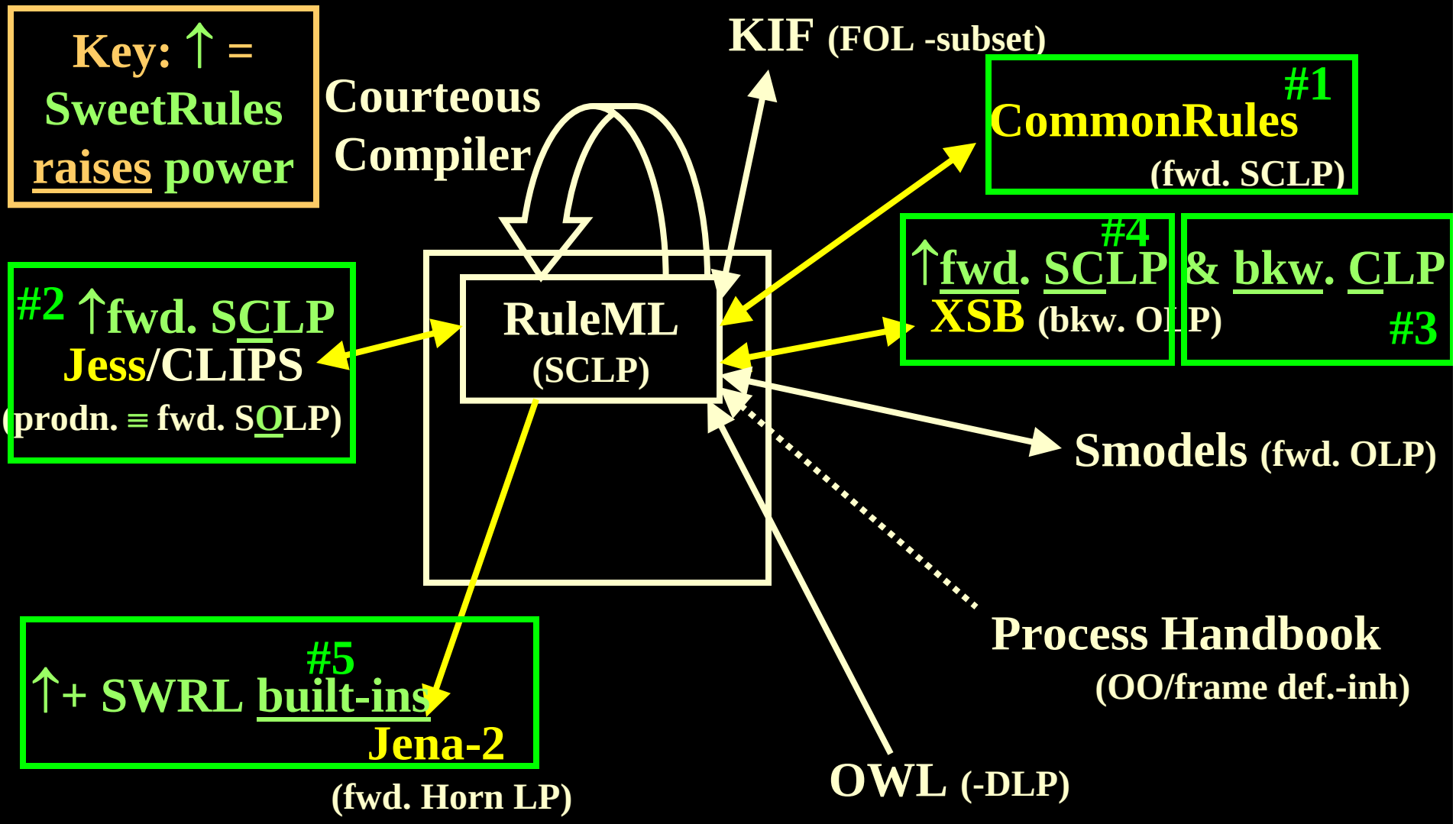
SweetRules V2.0 Fundamental KR

- Fundamental KR: Situated Courteous Logic Programs (SCLP) KR = Knowledge Representation
  - Horn
  - + Negation-As-Failure (NAF) = Ordinary LP
  - + Courteous prioritized conflict handling
    - overrides relation on rule labels, classical negation, mutex integrity constraints
  - + Situated sensing & effecting
    - Invoke external procedural attachments
    - Sensing = tests/queries; e.g., built-ins
    - Effecting = side-effectful actions, triggered by conclusions

SweetRules V2.0 Translators Graph



OPTIONAL: SweetRules V2.0 New Inferencing Engines



Objectives for Integrating Distributed SW Rules and Ontologies, Motivating SweetRules and its underlying theory+standards

BEFORE

AFTER

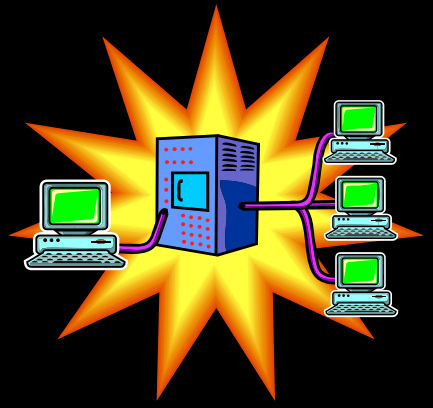
Contradictory conflict is globally contagious, invalidates all results.

⇒

Contradictory conflict is contained locally, indeed tamed to aid modularity.

Knowledge integration tackling the 5 D's (diversity, distributedness, disagreement, dynamism, & delay) is labor-intensive, slow, costly.

⇒



Knowledge integration is highly automated, faster, cheaper.

SweetRules: Use Cases Overview

- Contracts/negotiation, advertising/discovery
  - E-procurement, E-selling
  - Pricing, terms & conditions, supplier qualification, …
- Monitoring:
  - Exception handling, e.g., of contract violations
    - Late delivery, refunds, cancellation, notifications
  - Notifications, personal messaging, and other workflow
- Trust Policies: authorization, confidentiality & privacy, security, access control
  - E.g., financial services, health care
    - *Extensive analysis of business case/value*
- Semantic mediation: rule-based ontology translation, context-based information integration

Outline of Session

- Intro: What is the Semantic Web
  - Knowledge Representation in XML; Agents; with Web Services
- Why it Matters for Business
  - Knowledge-based Services Engineering
  - Examples of Policies for Contracting and Authorization
    - Pricing, Comparison Shopping, Ordering Lead Time, …
- Semantic Rules: Technology and Standardization
  - RuleML, Theory Advances, SweetRules Open Source Platform
- Roadmapping Business Value and Market Evolution
  - Cheaper, Faster, Better; EAI and B2B; Early Verticals
- Future Directions & Opportunities pertinent to ASR/SSME

Advantages of Standardized SW Rules for Policies, e.g., Authorization/Security

- Easier Integration: with rest of business policies and applications, business partners, mergers & acquisitions
  - Enterprise integration, B2B
- Familiarity, training
- Easier to understand and modify by humansChange management
- Quality and Transparency of implementation in enforcement
  - Provable guarantees of behavior of implementation
- Reduced Vendor Lock-in
- Expressive power
  - Principled handling of conflict, negation, priorities
- $\Rightarrow$ Agility, change management $\uparrow$

Advantages of SW Rules, cont'd:
Loci of Business Value in Policy Management

- Reduced system dev./maint./training costs
- Better/faster/cheaper policy admin.
- Interoperability, flexibility and re-use benefits
- Greater visibility into enterprise policy implementation $\Rightarrow$ better compliance
- Centralized ownership and improved governance by Senior Management
- Rich, expressive policy management language allows better conflict handling in policy-driven decisions
- Strategic agility, incl. wrt business model

SWS Adoption Roadmap: Some Strategy Considerations

- "Death. Taxes. Integration."
- 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
- Agility/speed/flexibility in business processes, supply chains
- "Killer app" target known for 30 years: do better job of EDI

Prospective SW Early Adopters: Areas by Industry or Task

- *We've discussed a number of industry or task areas:*
  - Manufacturing supply chain, procurement, pricing, selling, e-tailing, financial/business reporting, authorization/security/access/privacy policies, health records, credit checking, banking, brokerage, contracts, advertising, …
- Others:
  - travel "agency", i.e.: tickets, packages
    - See Trading Agent Competition, [M.Y. Kabbaj thesis]
  - military intelligence (e.g., funded DAML)

Outline of Session

- Intro: What is the Semantic Web
  - Knowledge Representation in XML; Agents; with Web Services
- Why it Matters for Business
  - Knowledge-based Services Engineering
  - Examples of Policies for Contracting and Authorization
    - Pricing, Comparison Shopping, Ordering Lead Time, …
- Semantic Rules: Technology and Standardization
  - RuleML, Theory Advances, SweetRules Open Source Platform
- Roadmapping Business Value and Market Evolution
  - Cheaper, Faster, Better; EAI and B2B; Early Verticals
- Future Directions & Opportunities pertinent to ASR/SSME

Our Contributions to Rule-based Semantic Web Services I

- Pioneered semantic rules core technology:
  - Knowledge representation theory, techniques, algorithms, standards design and standards momentum
    - Choice of declarative LP for interoperability among commercially important rule systems
    - Webizing in XML; URI-reference; RuleML standards design and effort
    - Situated LP, Courteous LP, Description LP, Production LP
  - Tools:
    - SweetRules: Showpiece of DARPA and RuleML
    - The earlier IBM CommonRules and IBM Agent Building Environment
      - Thousands of downloads
  - Applications: ... [see next slide]
  - Strong impact on SW architecture incl. W3C RIF launch

Our Contributions to Rule-based Semantic Web Services II

- Pioneered rule-based SWS application scenarios and prototypes in:
  - End-to-end contracting (e-commerce lifecyle), e.g., e-tailing, SCM
    - Includes application-specific techniques, with high knowledge reuse: SweetDeal approach
  - Business policies, trust authorization
  - In tandem with the core technology for semantic rules

- Co-led major portions of SWSF design
  - Led SWS Task Cluster Analysis and co-led Contracts/Negotiation area
    - Based largely on our SweetDeal and other applications work
  - Co-led Rules area, led convergence of SWSL to/with RuleML
  - Helped shape community concept of SWS

Our Contributions to Rule-based Semantic Web Services III

- Information integration, e.g., ontological context translation applied to financial reporting using ECOIN tool
  - Best paper award in the top management-academe IT conference (WITS '02, '04)
- Business value and roadmapping analysis
  - Pioneered it in/for the SWS community

Outline of Session

- Intro: What is the Semantic Web
  - Knowledge Representation in XML; Agents; with Web Services
- Why it Matters for Business
  - Knowledge-based Services Engineering
  - Examples of Policies for Contracting and Authorization
    - Pricing, Comparison Shopping, Ordering Lead Time, …
- Semantic Rules: Technology and Standardization
  - RuleML, Theory Advances, SweetRules Open Source Platform
- Roadmapping Business Value and Market Evolution
  - Cheaper, Faster, Better; EAI and B2B; Early Verticals
- Future Directions & Opportunities pertinent to ASR/SSME

Future Directions & Opportunities pertinent to ASR & SSME I

- Services science (SSME) vision generally
  - Automate deeply and cheaply
    - Raise IGS profit margins
  - Semantic integration: develop, leverage
    - Its criticality now more recognized in data management and Web Services communities
      - not just in SW community
  - Knowledge-based business processes
    - Across the firm
    - Across the market
    - Across the society
    - *Of course a lot of this is longer-term*

ASR= Almaden Services Research

SSME = Services Science, Engineering, and Management

Future Directions & Opportunities pertinent to ASR & SSME II

- Service descriptions
  - Representing them, e.g., with rules or FOL
    - Ontologies, including specific business domains
    - Pre- and post- conditions
    - Contractual aspects
    - Trust aspects
    - Exception handling, monitoring aspects
  - Using them to improve:
    - asset reuse
    - service composition and development
    - strategic outsourcing
  - Knowledge capture
    - Facilitating it, incenting it

Future Directions & Opportunities pertinent to ASR & SSME III

- Service contracts, policies, and trust
  - e.g., for SOA and WS
- Contracts about whatever, that invoke/specify services
  - E.g., VCR contract's refund or repair-shipment provisions
- Extend business informatics methodologies, e.g.:
  - Monitoring
  - Protocols
  - Component Business Model

Future Directions & Opportunities pertinent to ASR & SSME IV

- Combine structured knowledge with unstructured knowledge, e.g., text:
  - Annotation, tags
    - categories/ontologies, context
      - semantic supercomputing
  - Probabilistic/fuzzy, induction/data-mining result
    - need hybrid fundamental KR
  - Trust:
    - access decisions
    - belief degree

Future Directions & Opportunities pertinent to ASR & SSME V

- Leverage intelligent information integration
  - Merge knowledge bases or databases
  - Context integration
  - E.g., finance, travel, biomedical

- Distributed rule-based services
  - push and pull, composed
  - event-driven, active/reactive
  - Fundamental theory (KR), techniques

Future Directions & Opportunities pertinent to ASR & SSME VI

- Bring in economics and management science perspectives and results to SSME, e.g.:
  - Incentives, game theory, business value analysis, operations management, marketing science, market dynamics, social welfare
- Form external collaborations on SSME with researchers from:
  - Academe, esp.: management, systems engineering, computer science
  - Industry, esp.: business partners, standards-participants

Future Directions & Opportunities pertinent to ASR & SSME VII

- Domains of business applications:
  - Advertising, shopping: targeting
  - Intranet and email: knowledge management
  - Financial services, e.g.: trading, reporting
  - Medical records, e.g.: billing, insurance
  - Trust authorization, e.g., enterprise asset access
  - Customer care
  - *(Your favorite goes here)*

Outline of Session

- Intro: What is the Semantic Web
  - Knowledge Representation in XML; Agents; with Web Services
- Why it Matters for Business
  - Knowledge-based Services Engineering
  - Examples of Policies for Contracting and Authorization
    - Pricing, Comparison Shopping, Ordering Lead Time, …
- Semantic Rules: Technology and Standardization
  - RuleML, Theory Advances, SweetRules Open Source Platform
- Roadmapping Business Value and Market Evolution
  - Cheaper, Faster, Better; EAI and B2B; Early Verticals
- Future Directions & Opportunities pertinent to ASR/SSME

OPTIONAL SLIDES FOLLOW

OPTIONAL IMPACTS SLIDES FOLLOW

Impacts of LP & RuleML Approach I

- Pioneered this, initially in CommonRules design and piloting
- Approach of Declarative Logic Programs (LP) cf. RuleML as focus for semantic web (SW) rules grows to be highly influential then dominant
- In DAML program (the leading US research effort on SW):
  - Focus on Rules (e.g., at twice-yearly F2F PI Meetings) rises steadily throughout the 5-year program. (Co-lead DAML Rules.)
  - RuleML gets official home in 2002
- W3C workshop then standards effort launch (RIF) in 2005
  - Largely based on RuleML approach and RuleML effort's recruitment of participants.
  - RIF has twice the participant turnout as OWL at same stage.

Impacts: LP & RuleML Approach II

- Semantic web rules annual scientific workshop colocated with every ISWC, matures to full conference (RuleML-2005).
  - Co-organized from inception, co-chaired 1st conference
- Collaboration/cooperation from the relevant major research/standards efforts:
  - DAML, SWSI, WSML, W3C, OMG, Oasis, ISWC, Joint Committee
- Conference tutorials at ISWC-2004, ISWC-2005, WWW-2006
- Highly cited (e.g., cites to DLP, cites by semantic web rules papers)

Impacts in Semantic Web Services I

- SWSF (= SWS Framework from SWSI):
  - The leading design for general SWS service descriptions and ontologies, follow-on to OWL-S
  - Co-editor on Rules, and on Contracts/Negotiation
    - Also co-chair of Industrial Partners Program
  - SWSF Language chosen to be/extend RuleML
  - Created the services-tasks vs. KR cluster-analysis that underlies the overall SWSF technical approach & agenda including the Language design

Impacts in Semantic Web Services II

- SweetDeal approach to e-contracting influential
  - Key use case for SWS incl. SWSI
  - Key use case for SW rules & SW architecture
    - Motivates and pilots URI reference approach of RuleML
    - Motivates and pilots DLP
- Helped shape the definition of SWS
  - Organized and chaired the first major scientific conference panel on SWS (WWW-2003)

Impacts in Tool Development I

- SweetRules V2 (2004-)
  - Open source, multi-institutional
  - A showpiece of RuleML
    - Reference implementations & use cases
  - A showpiece of DARPA
    - The largest agenda item at final DAML F2F PI Meeting
    - Backstopped standardization push to W3C by DARPA

Impacts in Tool Development II

- IBM CommonRules (CR) (1999-)
  - Highly influential → RuleML
  - Thousands of downloads
    - Still getting revenue ☺
  - Piloted commercially in EECOMS
    - $29 Million supply chain effort by IBM-led industry consortium (2000)
    - Technical co-lead for EECOMS overall.

Impacts in Tool Development III

- IBM Agent Building Environment (ABE) (1996-1997)
  - Used in IGS personalized e-shopping offering and customer applications [business units that later became part of IGS]
  - Thousands of downloads. Influential in mindshare of intelligent agents (IA) R&D community.
  - Helped pioneer AlphaWorks (one of the first releases ever on it)
    - Largest focus of IBM company-wide IA activities in mid-1990's
  - Inspired IBM's ABLE as a follow-on (L = Learning)

Tool Development IV

- For each tool development project:
  - Created and led project and designs
  - Rapid development, quickening with experience
    - 6 months to 1st release of SweetRules
    - 12 months coding to 1st release of CommonRules
    - 18 months coding to 1st release of ABE, including transfer to remote Development organization
  - Moderate budget (e.g., shoestring <$400K for SweetRules)
  - On-time delivery:
    - ABE to IBM Software Group [business unit that later became part of SWG]
    - CommonRules to EECOMS IBM-led industry consortium
    - SweetRules to DARPA

Tandem Methodology

- Tandem methodology is to co-develop:
  1. core techniques/technology
  2. application scenarios, requirements, and prototypes
  - Dialectic of mutual feedback (1.) ↔ (2.)
  - Applications → requirements → drive development of core techniques
  - Core techniques/tools → piloted in application scenarios/prototypes

Impacts of Tandem Methodology I

- URI-reference and DLP techniques for combining rules with ontologies
  - in tandem with SweetDeal use cases, prototypes
    - E.g., RuleML rule condition uses an OWL class
- Courteous LP – incl. its mutex feature
  - in tandem with e-commerce retailing / e-contracts scenarios
    - E.g., uniqueness of price per item
- CommonRules and BRML/RuleML, incl. Courteous LP features
  - in tandem with EECOMS negotiation and exception handling

Impacts of Tandem Methodology II

- ABE, Situated LP
  - in tandem with personal messaging, mail, news, customer service info flow
- ECOIN information integration approach: rule-based mapping of ontological contexts
  - in tandem with financial reporting, travel agency
  - Influential in WITS community (premier venue for management-school academic IT researchers)
    - Best paper (2002), best paper nomination (2004) for follow-on

Impacts: Courteous LP

- An extension feature for LP
  - Enables robust, natural, scaleable merging/updating
- Adopted by RuleML and SWSF
- A focus in SW rules theoretical research community (e.g., RuleML conference)
- Growingly influential

Impacts: Translation of LP/RuleML ↔ Production Rules

- Part of what's now called Production LP
- Uses Situated LP for actions and tests
  - Situated LP an extension feature for LP, adopted by RuleML

- A focus in SW rules R&D community; many implementing translation of LP → Jess
- Underpins agenda/optimism/energy in W3C RIF that production rule vendors join the SW

Impacts: Strategy Activities

- Industry-wide/community-wide strategy co-leader
  - RuleML
  - SWSF

- Company-wide strategy technical co-lead on intelligent agents and business rules at IBM
  - White paper, evangelism in 1994
  - → non-Research development commitment on intelligent agents, with info-item-flow and e-commerce focus, in 1994-97

OPTIONAL SLIDES FOLLOW

For More Info: ISWC-2005 Tutorial

http://ebusiness.mit.edu/bgrosof/#ISWC2005RulesTutorial

E.g., Part C. Applications -- Policies, Services, and Semantic Integration

- 1. Ontology Translation and Semantic Integration
- - SWRL uses, ECOIN, financial services
- 2. End-to-End E-Contracting and Business Process Automation
- - supply chain, e-tailing, auctions, SweetDeal, Process Handbook
- 3. Business Policies including Trust (authorization, security, privacy, confidentiality, access control)
- - credit, health, RBAC & XACML access control, P3P privacy
- 4. Semantic Web Services
- - SWSL tasks
- 5. Prospective Early Adopter areas, strategy, and market evolution

Some Technical Directions for Research

- Incremental Reasoning: Events, Updates
- LP KR other extensions:
  - Existentials via skolemization
  - Combine Hilog higher-order features reducible to first-order; OWL-Full, RDF-Full
  - Equality: user-defined, nonmonotonic
  - Reification
- Hypermonotonicity: analysis of LP, merging; new KR's incl. disjunctive
- Probabilistic, decision-theoretic, game-theoretic; Inductive, learning, data mining
- Constraints: satisfaction, optimization

- Trust policies for firewalls, confidentiality, security, privacy, access control
- E-Contracting end-to-end reuse, power: incl. business process monitoring
- Policy Ontology, Services Ontologies, Relationship to C++/Java/C# Inheritance
- Web Services "Policy Management", "Contracts"
- Add semantics to existing standards: XBRL, XACML, ebXML, RosettaNet, EDI
- Biomedical: patient records privacy and workflow, drug discovery, treatment safety tracking
- Marketing, intelligence, supply chain, financial reporting, travel
- Business Value Analysis, Strategy, Roadmapping

Exploding Research Interest in SW

Since 2002: …

- International Semantic Web Conference (ISWC) formed
  - Grown to 400+ researchers
- Became 2nd largest topic area of the International Conference on the World Wide Web (WWW)
  - (1st is Search, i.e., Google etc.)
- Specialized conferences formed: e.g., RuleML
- Major Research Programs in US and EU
- Professional Societies Chapters formed: e.g., AIS SIG
- Journals formed: e.g., J. Web Semantics
- Several industry standards efforts (some done)

OPTIONAL SLIDES FOLLOW

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

More about Approach

- Many contract provisions well represented using rules
- Rule language alone is not enough: need communication infrastructure
- approach: communicate rules machine-to-machine/app-to-app using
- and new generation web infrastructure:
  - XML, "Semantic Web" and "Web Services" standards
  - + rule/knowledge engines and editors for developers
- XML messages used for interoperability
- XML aids wrappering and transformation of info between applications/organizations
- communal agreements on schemas and ontologies
- context mappings

Overview of RuleML Today III

- Logic Programs is a Fully Declarative KR (not simply Prolog!)
  - Well-established logic with model theory
  - Available algorithms, implementations
  - Close connection to relational DB's
    - core SQL is Datalog Horn LP
- Abstract graph syntax
  - 1st encoded in XML…
  - … then RDF … also a presentation syntax for human editing
- Expressive Extensions incrementally, esp. already:
  - Non-monotonicity: Negation as failure; Courteous priorities
  - Procedural Attachments: Situated actions/effecting, tests/sensing
  - Hilog, frame syntax,
  - *In-progress:*
    - reification cf. F-Logic Programs, SWSL
    - Events cf. Event-Condition-Action