SILK: Higher Level Rules with Defaults and Semantic Scalability

Benjamin Grosof*

October 26, 2009

1.5-Hour presentation at RR 2009**

* Vulcan Inc., benjaming@vulcan.com

http://www.mit.edu/~bgrosof/

http://silk.semwebcentral.org

** 3rd International Conference on Web Reasoning and Rules, Chantilly, VA, USA

http://www.rr-conference.org/RR2009/

PROJECT HALO

VULCAN

Outline of Talk

- Overview of SILK effort
  - Vision, Origins in parent Project Halo, Goals
  - Effort, Requirements, Approach, Roots, Status, Plans
  - Prototype, Theory, Language
- Examples and Use Cases: E-Science, E-Commerce, Trust, …
- Hyper Logic Programs KR approach and expressive features
  - More details on Requirements and Design
  - Higher-Order Defaults
    - Argumentation Theories approach (LPDA) to Defeasibility
  - Remedying FOL Semantics' Lack of Scalability
    - Weakened Classical, via Hypermonotonic mapping.
  - Comparison to other semantic rule systems and standards
    - RIF, BRMS, OWL, DBMS, etc.
- Conclusions and Directions
  - Roadmap for SILK and Industry
  - How You can be Involved

PROJECT HALO

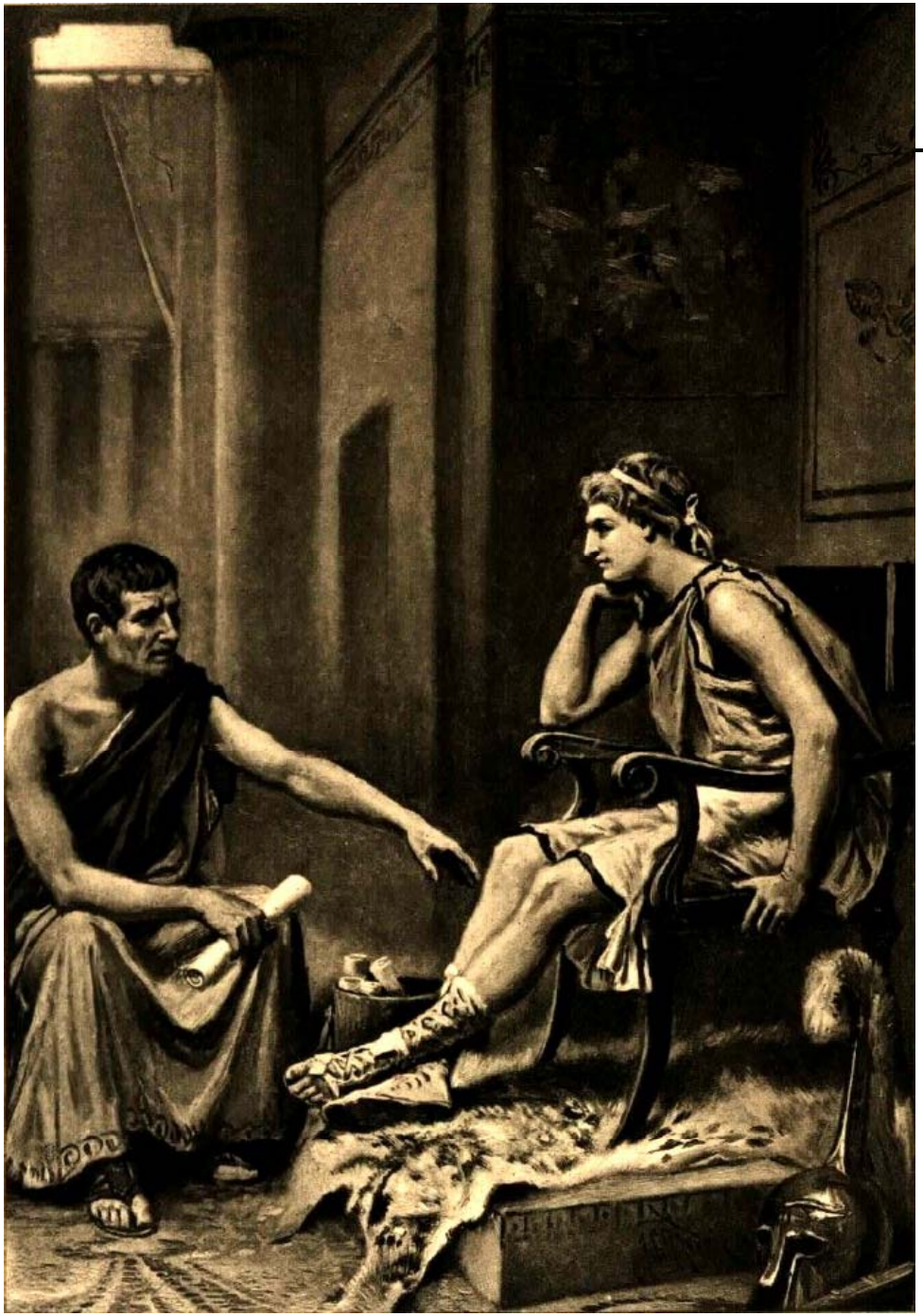
VULCAN

Vulcan's Project Halo Begins; 1st system is AURA

- Vision of Digital Aristotle
  - Put the bulk of the world's scientific and similar knowledge on-line
  - Answer questions, act as personal tutor, with deep reasoning
- How to operationalize Digital Aristotle as a research effort?
- College-level science selected as initial domain focus
  - Medium wide, medium deep.
  - Good metrics available: textbook-type exam Q's. Initial domain task focus is:
    - Advanced Placement Exam (AP) in Physics, Chemistry, and Biology
      - Taken by USA high-school students to get credit for 1st-year college courses
- AURA expert system developed (see http://www.ai.sri.com/project/aura)
  - Novel combination of available techniques from AI
  - Controlled Natural Language, GUI, Frame-based KR, Problem-Solving
  - Students as users – formulate questions, formulate knowledge
  - Initial version 2004, then refined extensively and tested rigorously

PROJECT HALO

VULCAN



Aristotle Tutoring Alexander

VULCAN

Halo Enters Semantic Web Era; 2nd system is SMW+

- How to enable effective Knowledge Acquisition (KA)?
  + By Subject Matter Experts (SMEs), not programmers or knowledge engineers
  + Collaboratively – incorporate large #s of SMEs in KB construction & maintenance
  + Leveraging the Web
- Halo Extension to Semantic MediaWiki (SMW+) developed.
  - Open source extension of the MediaWiki software Wikipedia runs on
  - Supports RDF and OWL subset, interleaved tightly with hypertext
  - Rapid maturation of initial functionality
  - Standing queries. Data import/export. Plug-ins.
  - Upcoming release: simple semantic rules (Horn) and access control
  - Strong community uptake, early commercial adoption already
  - For more, see http://wiki.ontoprise.de
- But need better KR too, in part for sake of KA.
  - The underlying KR is the target for KA: "The KR is the deep UI"
  - Web knowledge interchange (with merging) for scalability of collaborative KA

PROJECT HALO

VULCAN

Goals for SILK KR Effort – Halo's 3rd system

- Expressiveness + Semantics + Scalability
  - Push the Frontier: high risk, high return
- Address requirements for AURA on AP task (& for SMW+)
  - Expressive power (e.g., defaults and processes)
  - Understandability via semantics and expressiveness
    - Raise abstraction level closer to the user's natural language and cognition
- Address requirements for long-term Digital Aristotle vision
  - Wider set of domains and tasks, via KR expressiveness and better KA
  - Knowledge interchange via semantics and expressiveness
  - Performance scalability of reasoning (incl. truth maintenance)

PROJECT HALO

VULCAN

Expressiveness "Brittleness" Areas Targeted

- Defaults/Exceptions/Defeasible *(incl. nonmonotonic reasoning, theory revision, argumentation, truth maintenance)*
  - A kinematics problem situation has standard earth gravity, and no air resistance. [physics AP]
  - A given organism has the anatomy/behavior that is typical/normal for its species, e.g., a bat has 2 wings and flies. [bio AP]
  - Price info for an airplane ticket on Alaska Air's website is accurate and up to date. [e-shopping]

  ❖ Practical reasoning almost always involves a potential for exceptions
- Hypotheticals
  - If Apollo astronaut Joe golfed a ball on the moon, then standard earth gravity would not apply. [negative hypothetical] [*conflict* between defaults, resolved by *priority* among them]
  - If I had swerved my car 5 seconds later than I did, I would have hit the debris in the left lane with my tire. [*counterfactual*]
- Actions and Causality
  - If a doorkey is incompletely inserted into the keyhole, turning the key will fail. [*precondition*]
  - During the mitotic stage of prometaphase, a cell's nuclear envelope fragments [biology AP]
  - After a customer submits an order on the website, Amazon will email a confirmation and ship the item. [Event-Condition-Action (*ECA*) rule] [policy]
- Processes (i.e., representing and reasoning about processes)
  - Mitosis has five stages; its successful completion results in two cells. [compose] [partial description]
  - If Amazon learns that it will take an unexpectedly long time to stock an ordered item, then it emails the customer and offers to cancel the order without penalty. [exception handling]
  - A Stillco sensor-based negative feedback thermal regulator is adequate to ensure the overnight vat fermentation of the apple mash will proceed within desired bounds of the alcohol concentration parameter. [science-based business process]

Ubiquitous in science, commonsense, business, etc. All are interrelated.

PROJECT HALO

VULCAN

SILK's ambitious Vision for longer-term Impact

- Key Knowledge Representation (KR) infrastructure sufficient to enable
  creation of global, widely-authored, very large knowledge bases (VLKBs)
  about science and business*
  that answer questions
  and proactively supply information,
  using powerful reasoning about rules and processes,
  that can be customized in their content and actions
  for individual organizations or people

- Newest part of Vulcan's Project Halo
  which addresses the problems of scale and brittleness in KBs,
  including the Knowledge Acquisition and UI aspects

* "Business" here is shorthand for human affairs, incl. government

PROJECT HALO

VULCAN

SILK & Hyper LP: Overview

- A KR Language and KR System with reasoner, UI, interchange
  - Syntax & semantics, forward & backward inferencing, Java API, translators
- Goal: Expressiveness + Semantics + Scalability + Web
- Focus: Defaults and Processes
- Largest rule research program in the US (that we are aware of)
  - Begun in 2008, part of Vulcan's Project Halo, primarily via contractors
- *Hyper* LP KR combines new features
  - Defaults and Weakened Classical, cf. generalized Courteous LP and Hypermon. map.
  - External Actions and Events and Queries, cf. generalized Production LP

  with previous advanced features
  - Higher-order and Frames, cf. Hilog and F-Logic
  - Webized syntax, cf. RIF/RuleML and OWL/RDF
  - Closed-World, cf. well-founded unstratified NAF
  - Good Efficiency of reasoner performance
  - Equality, Functions, and misc. other less glamorous features
- Status: prototype engine, language spec, and theory for expressive core
  - V1 adds Higher-Order Defaults to FLORA-2
  - Extensive requirements analysis, use cases, benchmarking; ReCyc translation
  - V2 in development adds more features and Java API (See ISWC-2009 Demo!)

PROJECT HALO

VULCAN

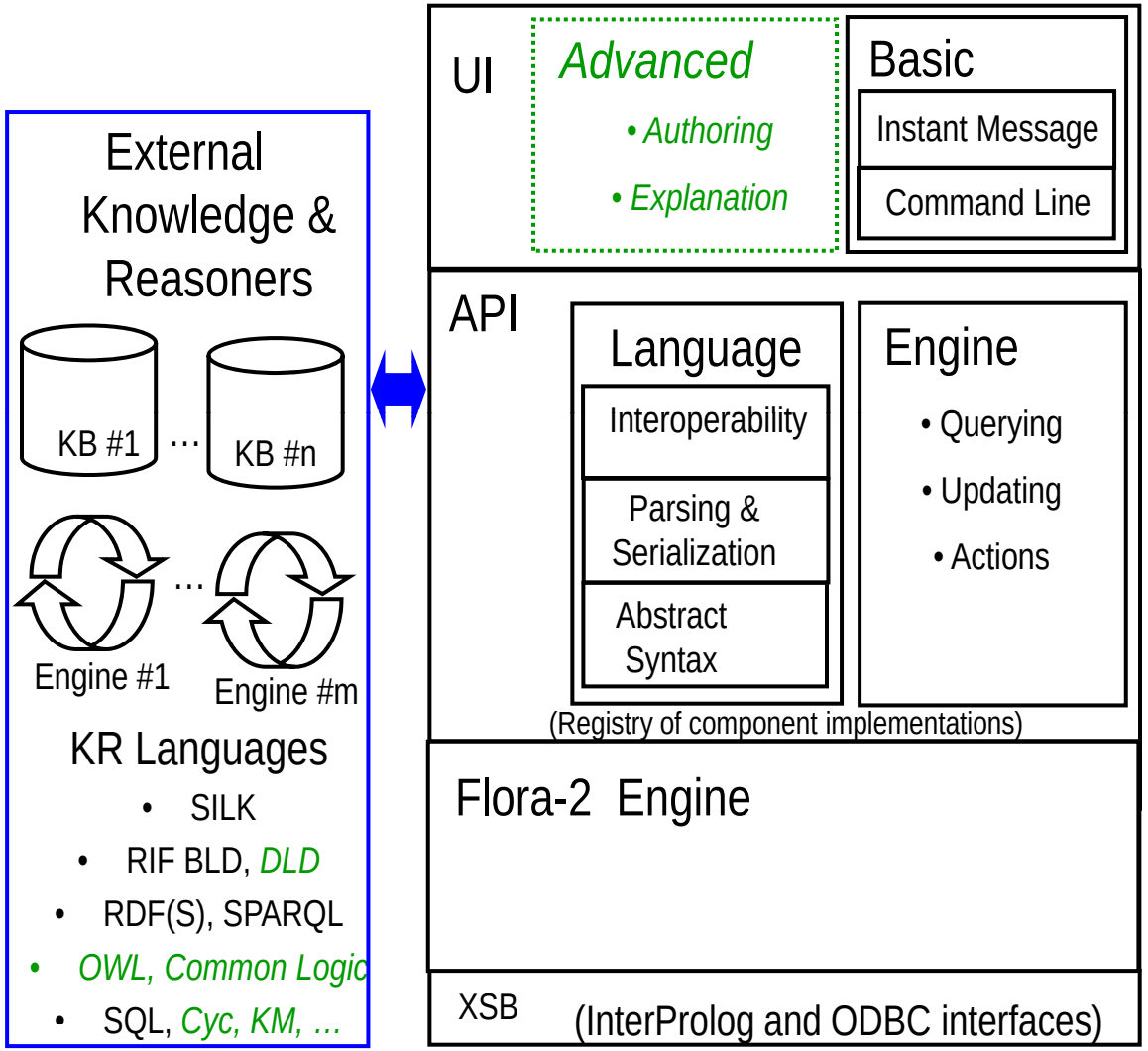
SILK & Hyper LP Overview (cont.'d)

- Radically extends the KR power of W3C OWL, SPARQL, and RIF –- and of SQL
  - Defaults and robust conflict handling – *cope with knowledge quality and context*
  - Higher-order and flexible meta-reasoning – *elevate meta-data to meta-knowledge*
  - Actions and events, cf. production rules and process models – *activate knowledge*
- Raises the KR abstraction level for business users (SMEs) and NL KA/UI
- Use cases in business policies, ontology mapping, e-commerce, biomed, ...
- Redefining the KR playing field for Semantic Web, business rules, and rule-based process management
  - Defaults and Higher-Order – yet retain computational web scalability
  - Escape from Glass Bubble– yet retain grade-AAA model-theoretic semantics
- Motto: "Transforming Knowledge"
  - Composes a set of KR transformations for …
  - Expressive extensions – language and semantics
  - Translations between KRs/syntaxes, for interchange
  - Reuse of previous algorithms and implementations
- http://silk.semwebcentral.org

PROJECT HALO

VULCAN

SILK V2 Architecture and V3 Directions



- V2 Functionality
  - Higher-order defaults reasoning, combines many other advanced KR features
  - SILK and external KR language support integrated tightly with reasoning engine
- *Future Items*
  - *Meet Process req.'s*
  - *More UI is key: graphical, limited NL*
  - *Integrate with AURA*
  - *SILK KR: truth maintenance, probabilistic & constraints, parallelization*
- Test Sets Focus
  - Defaults, Process
  - AP esp. Bio

NB: *Italics* indicate future items beyond V2

Hyper Logic Programs: Other Features

- HLP combines further a number of other extensions of LP, notably:
- Webizing, cf. RuleML and RIF
  - URIs for predicates and other logical constants
  - Load-time import of knowledge bases over the Web
- External Queries and Actions, cf. Production LP (and Situated LP)
  - Via procedural attachments. Including built-ins.
  - Enables interoperation with Production/ECA rules (via SweetRules technique)
  - Brings Actions (and events) to the semantic party
- External Events, via newly modified approach
- Equality, incl. explicit derived, via newly modified approach
- Lloyd-Topor, Aggregations, Integrity Constraints, Skolemization, Functions, misc. other features
- *HLP is still under development (there's a lot of new expressiveness)*

PROJECT HALO

VULCAN

SILK Current Status – More

- New approach to representing causal change in processes
  - Uses defaults
- Use cases, incl. survey
  - Science AP
  - Business domains
- ReCyc: Rough prototype translator from Cyc to SILK
  - 3 Million axioms from ResearchCyc (translates 99% of the KB)
- Benchmarking of relevant rule systems
  - OpenRuleBench [Liang *et al.* WWW-2009]
- SILK V2 is in development
  - Currently alpha with most functionality but needs general polishing
  - Upcoming conference demos at ISWC-2009 and RuleML-2009

PROJECT HALO

VULCAN

SILK Contributors current/past (partial list)

- Vulcan (Benjamin Grosof, Mark Greaves, Dave Gunning)
- Stony Brook University (Michael Kifer; students H. Wan, S. Liang, P. Fodor)
- SRI International (Vinay Chaudhri, David Martin, Ken Murray, Bill Jarrold)
- BBN Technologies (Mike Dean, Dave Kolas, Matt Rubin)
- Ontoprise GmbH (Daniel Hansch, Jurgen Angele)
- Automata (Paul Haley)
- Boeing (Peter Clark)
- Cycorp (Keith Goolsbey, Doug Lenat, Ben Rode)
- University of Texas (Bruce Porter, Ken Barker)
- University of Toronto (Sheila McIlraith; students S. Sohrabi, H. Ghaderi)
- University of Amsterdam (Bert Bredeweg)
- University of Freiburg (Georg Lausen)
- University of Michigan (Michael Wellman)
- Richard Fikes, consultant (Stanford University)
- Raphael Volz, consultant

SILK-relevant Cooperations (partial list)

- Project Halo has cooperations with other major research efforts:
- LarKC (The Large Knowledge Collider), funded by EU
  - http://www.larkc.eu
- NeOn (Lifecycle Support for Networked Ontologies), funded by EU
  - http://www.neon-project.org
- DARPA

PROJECT HALO

VULCAN

Outline of Talk

- Overview of SILK effort
  - Vision, Origins in parent Project Halo, Goals
  - Effort, Requirements, Approach, Roots, Status, Plans
  - Prototype, Theory, Language
- Examples and Use Cases: E-Science, E-Commerce, Trust, …
- Hyper Logic Programs KR approach and expressive features
  - More details on Requirements and Design
  - Higher-Order Defaults
    - Argumentation Theories approach (LPDA) to Defeasibility
  - Remedying FOL Semantics' Lack of Scalability
    - Weakened Classical, via Hypermonotonic mapping.
  - Comparison to other semantic rule systems and standards
    - RIF, BRMS, OWL, DBMS, etc.
- Conclusions and Directions
  - Roadmap for SILK and Industry
  - How You can be Involved

PROJECT HALO

VULCAN

Ecology Ex. of Causal Process Reasoning in SILK

```
/*    Toxic discharge into a river causes fish die-off.    */
/* Init. facts, and an "exclusion" constraint that fish count has a unique value */
  occupies(trout,Squamish).
  fishCount(s0,Squamish,trout,400).
  silk:opposes(fishCount(?s,?r,?f,?C1), fishCount(?s,?r,?f,?C2)  :-  ?C1 != ?C2.
/* Action/event description that specifies causal change, i.e., effect on next state */
  @tdf1  fishCount(?s+1,?r,?f,0) :- occurs(?s,toxicDischarge,?r) and occupies(?f,?r).
/* Persistence ("frame") axiom */
  @pef1  fishCount(?s+1,?r,?f,?p) :- fishCount(?s,?r,?f,?p).
/* Action effect axiom has higher priority than persistence axiom */
  @pr1  silk:overrides(tdf1,pef1).
/* An action instance occurs */
  @UhOh  occurs(s0+1,toxicDischarge,Squamish).
```

As desired: |= fishCount(s0+1,Squamish,trout,400) and
fishCount(s0+2,Squamish,trout,0).

Notes: @ prefixes a rule label. ? prefixes a variable. :- means if. Opposes indicates an exclusion constraint between two literals, which means "it's a conflict if"..

E-Commerce Ex. of Causal Process Reasoning

```
/*    E-commerce delivery logistics. */
/* Initial fact, and prevention constraint that location is unique */
  loc(s0,PlasmaTV46,WH_LasVegasNV);
  silk:opposes(loc(?s,?item,?posn1), loc(?s,?item,?posn2))   :-   ?posn1 != ?posn2;
/* Action/event description that specifies causal change, i.e., effect on next state */
  @{mov1}  loc(?s+1,?item,?addr) and neg loc(?s+1,?item,?warehouse) :-
        shipment(?s,?item,?warehouse,?addr) and loc(?s,?item,?warehouse);
/* Persistence ("frame") axioms about location */
  @{pel}  loc(?s+1,?item,?posn) :- loc(?s,?item,?posn);
  @{pel}  neg loc(?s+1,?item,?posn) :- neg loc(?s,?item,?posn);
/* Action effect axiom has higher priority than the persistence axioms */
  silk:overrides(mov1,pell.
/* An action instance occurs */
  @{deliv57}  shipment(s0+1, PlasmaTV46, WH_LasVegasNV, Nine_Fog_St_SeattleWA);
```

As desired: |= loc(s0+2, PlasmaTV46, Nine_Fog_St_SeattleWA);

|= neg loc(s0+2, PlasmaTV46, WH_LasVegasNV);

PROJECT HALO

VULCAN

Trust Mgmt. Ex. of Higher-Order Defaults in SILK

illustrating also basic Knowledge-level Communication, and Frame syntax

In Frame syntax: subject[property -> object] *stands for* property(subject,object).

```
/*    Trust policy administration by multiple agents, about user permissions */
/* Admin. Bob controls printing privileges including revocation (neg). */
  Bob[controls -> print].   Bob[controls -> neg print];   /* neg print means it's disallowed.*/
  Cara[controls -> ?priv];  /* Cara is the most senior admin., so controls all privileges. */
/* If an administrator controls a privilege and states at a time (t) that a user has a privilege,
   then the user is granted that privilege. Observe that ?priv is a higher-order variable. */
  @grant(?t) ?priv(?user) :- ?admin[states(?t) -> ?priv(?user)] and ?admin[controls(?priv)];
/* More recent statements have higher priority, in case of conflict.  */
  overrides(grant(?t2), grant(?t1)) :- ?t2 > ?t1 ;
/* Admin.'s Bob and Cara make conflicting statements over time about Al's printing */
  Cara[states(2007) -> print(Al)].   Cara[states(2007) -> webPage(Al)] ;
   Bob[states(2008) -> neg print(Al)] ;
```

As desired: |= neg print(Al) and webPage(Al).

/* Currently, Al is permitted a webpage but not to print. */

PROJECT HALO

VULCAN

SILK Roots

- SILK draws upon previous work on semantic rules
  - W3C Rule Interchange Format (RIF)
  - RuleML incl. SWRL
  - SWSL (Semantic Web Services Lang.) and WSML
  - Flora and XSB, SweetRules, DLV
  - IBM Common Rules, Ontoprise Ontobroker
  - Description LP, W3C OWL 2 RL, Oracle SW rules
  - OMG PRR
  - ISO Common Logic and OMG SBVR
  - Jena, cwm and N3
  - SQL, SPARQL, XQuery
  - Theory and algorithms of KR from LP, AI, and DB communities

PROJECT HALO

VULCAN

Use Cases for SILK beyond commercial state of art

- There are many!
- Existing use cases from SILK's research-y or standards-design roots
  - E.g., from RIF, RuleML, SWSL documents and prototypes
  - E-commerce, financial, health, trust, SOA, policies, regulations, mobile, biomed, defense, etc.
  - Many of these are not yet implementable in current well-supported, well-performing commercially deployed systems
    - E.g., they use defaults
    - E.g., they use feature combinations that are not easily available

PROJECT HALO

VULCAN

Outline of Talk

- Overview of SILK effort
  - Vision, Origins in parent Project Halo, Goals
  - Effort, Requirements, Approach, Roots, Status, Plans
  - Prototype, Theory, Language
- Examples and Use Cases: E-Science, E-Commerce, Trust, …
- Hyper Logic Programs KR approach and expressive features
  - More details on Requirements and Design
  - Higher-Order Defaults
    - Argumentation Theories approach (LPDA) to Defeasibility
  - Remedying FOL Semantics' Lack of Scalability
    - Weakened Classical, via Hypermonotonic mapping.
  - Comparison to other semantic rule systems and standards
    - RIF, BRMS, OWL, DBMS, etc.
- Conclusions and Directions
  - Roadmap for SILK and Industry
  - How You can be Involved

PROJECT HALO

VULCAN

LP is the Core KR in today's world ... including the Semantic Web

- LP is the core KR of structured knowledge management today
  - Databases
    - Relational / SQL
    - XML semi-structured / XQuery
    - RDF semi-structured / SPARQL (triple stores)
  - Semantic Rule Standards
    - RuleML standards design
    - Rule Interchange Format (RIF)*
  - Semantic Ontologies
    - Most commercial implementations of OWL & RDFS are based on semantic rules: Description Logic Programs (DLP) + moderate extensions. E.g., Oracle.
    - OWL 2* includes the RL Profile, i.e., its Rules subset
- The Semantic Web today is mainly based on LP KR
  - … and thus essentially equivalent to semantic rules
  - You might not have realized that!

PROJECT HALO

VULCAN

Sem Tech Industry Requirements targeted by SILK

- Need to raise abstraction level, e.g., for SME and NL KA/UI
  - (SME = Subject Matter Expert, a.k.a. Business User)
- Need robustness & meta-reasoning for web KB integration
  - Cope with conflict, mediation, context, knowledge quality
  - Defaults ⇒ robustness, modularity ⇒ scalability
  - Higher-order ⇒ puts the meta- deeply in knowledge not just data
- Hope: be like advance of the Relational model in DBMS
  - Will Hyper LP be to the 2010s what Relational was to 1970s-80s?
    - (NB: software industry clockspeed was slower back then)

PROJECT HALO

VULCAN

Strategy on Expressiveness

- Start from declarative LP
  - Semantics available, but enables nonmonotonicity, unlike classical
  - A multitude of small and large expressive extensions available
    - Can hope to combine defaults with most of the other major ones

- Opportunity: newly combine tightly and synergize several major strands of pure-research progress in logical KR based on extensions of LP from the last 20 years
  - Good stuff, but pieces on the floor

- Build up expressiveness in layers (and by relaxing restrictions)
  - Extend syntax and semantics as we go

PROJECT HALO

VULCAN

More KR Rationale about SILK & Hyper LP

- "*Hyper*" since it is Web (*hyper*text) centric, and it behaves *hyper*monotonically
- It integrates several major LP extensions never previously combined:
  - Higher-order and Frames and Skolemization, cf. F-Logic
  - + Defaults, cf. Courteous LP (and Defeasible Logic)
    - Newly generalized and modified approach using Argumentation Theories
    - Sound Interchange with Full Classical Logic, via Hypermonotonic mapping
      - Unrestricted clauses, plus skolemization: greatly generalizes DLP, OWL RL
      - Behaves robustly in the face of knowledge quality errors and conflicting merging
    - Pervasively combined with all other KR features
- Give up reasoning by cases
  - Source of exponential worst-case complexity in classical, disjunctive LP, stable LP
  - Can hope to reintroduce in restricted or altered form, or develop work-arounds, later
  - But there are many apps not requiring it, e.g., DBMS, BRMS
- Can realistically hope to be web-scalable performance-wise, unlike highly expressive classical or answer set programs
  - Polynomial computational complexity, under non-onerous restrictions
    - Same complexity as Horn rules!! (Must be careful of recursion through functions.)
  - Many optimizations available
  - Established track record of high scalability for relational databases

PROJECT HALO

VULCAN

Driving Requirements for SILK Expressiveness

- Processes *[For science, BPM. E.g., >50% of questions on Environmental Sci. AP.]*
  - Actions, Causality, Events, Reactivity, State Change
- Knowledge-level Communication *[Knowledge, science, & business are societal]*
  - I.e., Import and Merge of External Knowledge, incl. data/facts, ontologies, rules
  - Via Pull/Query, and Via Push/Events
  - From Web, built-ins, specialized reasoners, broad-purpose reasoners
  - **Mediate** ontologies and contexts
  - Interchange with **Classical** logic KR, as well as with LP/rules KR
  - *Uses for Classical include:*
    - *Background KBs, e.g., ontology, e.g., about processes*
    - *Existing techniques and KBs for equations, constraints, and processes*
    - *Common Logic (and KIF), SBVR, OWL, RDF*

PROJECT HALO

VULCAN

Uses of Major SILK Expressive Features

- Defaults (beyond naf) *[For many purposes, pervasively]*
  - Exceptions, Priorities, Inheritance, Strong Negation, Preventive Integrity Constraints
  - *For OO, robust KB merging/updating, process causality, policy and regulation/law, natural language incl. KA, import of classical, argumentation, hypotheticals and counterfactuals*

- Higher-order, incl. for Meta-reasoning *[For many purposes, pervasively]*
  - *Convenient, concise abstraction for KR designers, and for KE/SME users*
  - *Many KRs have some of it, incl. RDF, OWL-Full, BRMS, Cyc. E.g., transitive_closure(?P).*
  - *Meta-reasoning uses include: KR macros, KB translation/import, ontology mappings, reasoning control, provenance, KB modularization, navigation in KA, multi-agent & nested belief, context, modals. Plus – the Web is about* meta-*data.*

PROJECT HALO

VULCAN

More SILK Expressive Requirements

- External Actions, Events, and Queries
  - Via **procedural attachments**. E.g., query **built-ins**.
  - Similar to production rules and Event-Condition-Action rules
  - *For knowledge communication and processes*
- Webized syntax
  - URI names for predicates, individuals, functions, KBs, and attached procedures
  - XML/RDF interchange format for the KR
  - *For knowledge communication*
- Equality (derived via non-fact rules) *[For entity identity and numerical reas.]*
  - Complex explicit derived equalities/equations. Inequalities too.
- Functions (logical) *[For higher-order and process recursion]*

PROJECT HALO

VULCAN

More SILK Expressive Requirements, continued

- Closed-World *[For defaults, numerical, collections, and meta-reasoning]*
  - Unstratified (not just stratified) negation-as-failure (NAF, a.k.a. “weak” negation)
    - Well-founded semantics for NAF so as to preserve tractability and well-definedness
  - Aggregate operators, e.g., count, total, average, setOf. NB: these depend on NAF.
  - Lloyd-Topor (freer appearance of logical connectives). NB: this depends on NAF.
    - {and, naf, or, exists, forall, implies} in body, {and, implies, forall} in head

- Frame syntax *[Convenient & familiar, e.g., RDF, OWL, UML, Aura]*
  - Frame (Object-Oriented style) syntax cf. F-Logic
- Skolemized existentials *[Convenient & familiar, e.g., RDF, OWL, UML, Aura]*
- Integrity constraints *[Convenient & familiar, e.g., DBMS, UML, Aura]*
  - Report violations
  - Prevent violations (via “exclusions”)

PROJECT HALO

VULCAN

Relevant Expressive Foundations – from our ISWC-2009 Tutorial

1. Overview of Logical Knowledge Representations
   - Logic Programs (LP) and its relationship to First Order Logic (FOL)
   - Rule-based Ontologies: Description Logic, Description LP, OWL RL
2. Basics: Horn Case; Functions
3. F-Logic, Frame Syntax, Object Oriented Style
4. HiLog, Higher-Order Syntax, Reification, Meta-Reasoning
5. W3C Rule Interchange Format: Dialects, Framework
6. Nonmonotonicity: Defaults, Negation, Priorities; FOL's Glass Bubble
   - Semantics for Default Negation
   - Courteous LP, Argumentation Theories
   - Hypermonotonic Mapping: FOL $\leftrightarrow$ LP, Soundly
7. Procedural Attachments to Actions, Queries, Built-ins, and Events
   - Production/Situated LP, Production Rules
8. Additional Features: Integrity Constraints, Inheritance, Lloyd-Topor, Equality, Skolemization, Aggregation, Datatypes, "Constraints"
9. Hyper LP and SILK – Putting it all together

PROJECT HALO

VULCAN

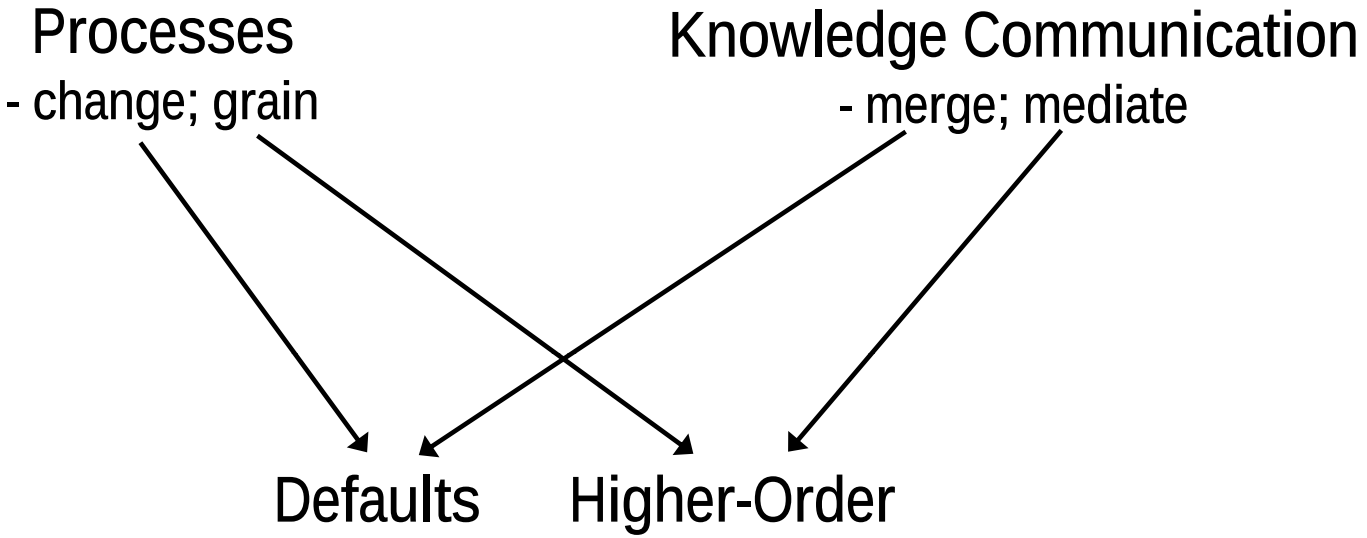
SILK Other Reasoning Requirements

- Explanations: to users and machines
- Performance Scalability of Inferencing
  - Exploit Parallelism
- Support Forward-Direction and Persistence in Inferencing
  - Persistent queries and conclusions
  - Truth Maintenance, handling nonmonotonicity and update/event flows
- Knowledge interchange, with translation between KRs/systems
  - Via Pull and Push, dynamically, over Web.
    - Data/Facts, Ontologies, Rules
  - Support relevant standards, therefore, e.g., RIF, OWL, RDF, Common Logic
  - Interoperate with Production Rules and similar Event-Condition-Action (ECA) rules
  - Trust management
- *Live in a Distributed World, generally*

PROJECT HALO

VULCAN

Dependencies among Requirements I

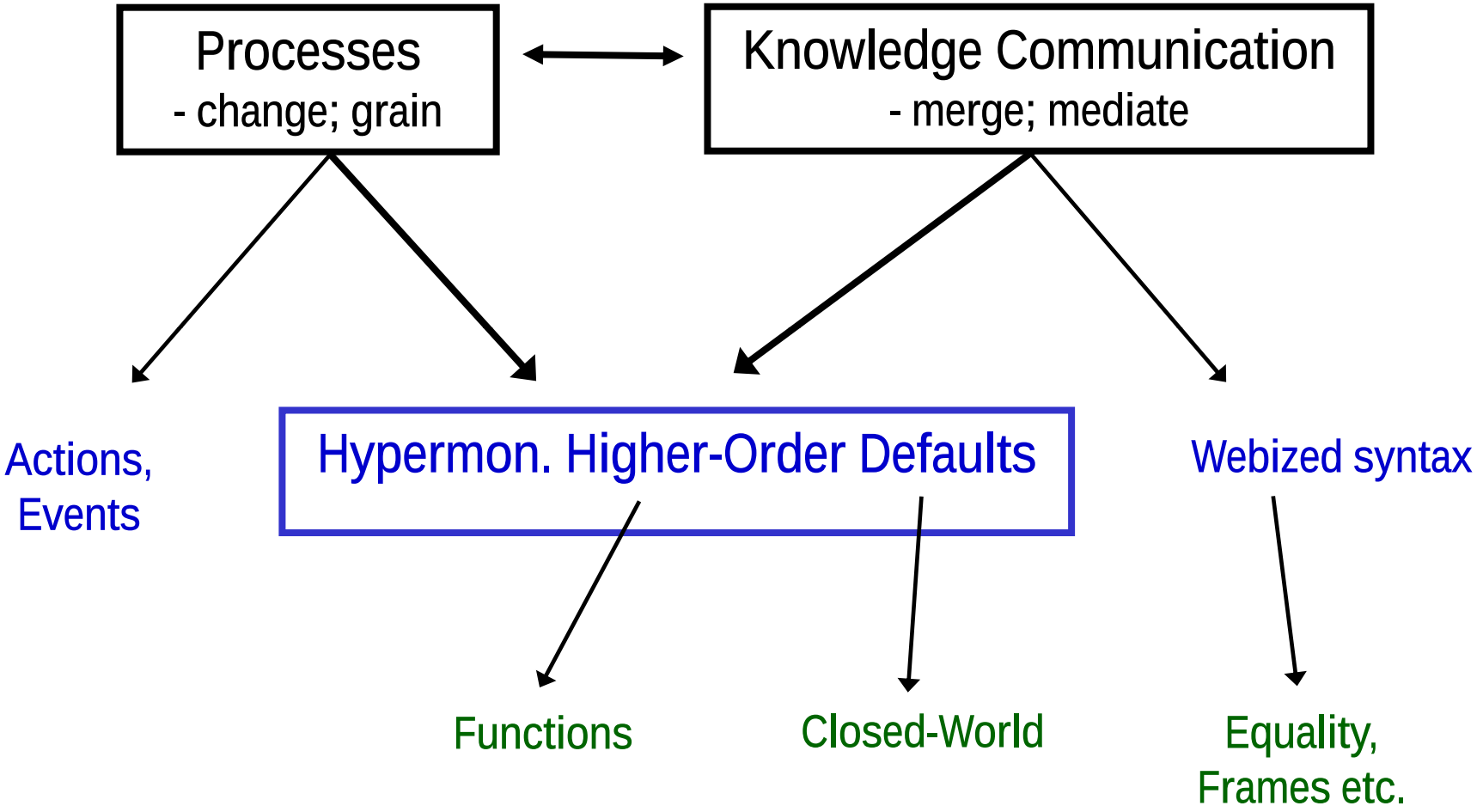


PROJECT HALO

VULCAN

33

Dependencies among Requirements II



PROJECT HALO

VULCAN

34

Outline of Talk

- Overview of SILK effort
  - Vision, Origins in parent Project Halo, Goals
  - Effort, Requirements, Approach, Roots, Status, Plans
  - Prototype, Theory, Language
- Examples and Use Cases: E-Science, E-Commerce, Trust, …
- Hyper Logic Programs KR approach and expressive features
  - More details on Requirements and Design
  - Higher-Order Defaults
    - Argumentation Theories approach (LPDA) to Defeasibility
  - Remedying FOL Semantics' Lack of Scalability
    - Weakened Classical, via Hypermonotonic mapping.
  - Comparison to other semantic rule systems and standards
    - RIF, BRMS, OWL, DBMS, etc.
- Conclusions and Directions
  - Roadmap for SILK and Industry
  - How You can be Involved

PROJECT HALO

VULCAN

Ubiquity of Priorities
in Commercially Important Rules -- and Ontologies

- Updating in relational databases
  - more recent fact *overrides* less recent fact
- Static rule ordering in Prolog
  - rule earlier in file *overrides* rule later in file
- Dynamic rule ordering in production rule systems (OPS5)
  - "meta-"rules can specify agenda of rule-firing sequence
- Event-Condition-Action rule systems rule ordering
  - often static or dynamic, in manner above
- Exceptions in default inheritance in object-oriented/frame systems
  - subclass's property value *overrides* superclass's property value, e.g., method redefinitions
- All lack Declarative KR Semantics

PROJECT HALO

VULCAN

Defeasible Reasoning

- Rules can be true by default but may be defeated
  - A form of commonsense reasoning
- Application domains:
  - policies, regulations, and law
  - actions, change, and process causality
  - Web services
  - inductive/scientific learning
  - natural language understanding
  - …
- Existing approaches:
  - Courteous Logic Programs (Grosof , 1997)
    - The main approach used commercially (IBM Common Rules, 1999)
  - Defeasible logic (Nute, 1994) *[similar to Courteous LP]*
  - "Prioritized defaults" (Gelfond & Son, 1997)
  - Preferred answer sets (Brewka & Eiter, 2000)
  - Compiling preferences (Delgrande et al., 2003)
  - …

PROJECT HALO

VULCAN

Semantical KR Approaches to Prioritized LP

The currently most important for Semantic Web are:

1. Courteous LP
   - KR extension to Ordinary LP
   - In RuleML, since 2001
   - Commercially implemented and applied
     - IBM CommonRules, since 1999
2. Defeasible Logic
   - Closely related to Courteous LP
     - Less general wrt typical patterns of prioritized conflict handling needed in e-business applications
     - In progress: theoretical unification with Courteous LP

PROJECT HALO

VULCAN

Courteous LP: the What

- Updating/merging of rule sets: is crucial, often generates conflict.
- Courteous LP's feature prioritized handling of conflicts.
- Specify scope of conflict via a set of exclusion constraints
  - Each is a preventive spirit integrity constraint on a set of competing literals
    - It says that not all of the competing literals can be entailed as true.
    - opposes(p, q) ≈ (⊥ :- p and q) // Case of 2 competing literals
  - opposes(discount(?product,"5%"), discount(?product,"10%"));
  - opposes(loyalCustomer(?cust,?store), premiereCustomer(?cust,?store));
- Permit strong negation of atoms: (NB: a.k.a. (quasi-) "classical" negation.)
  - ¬p means p has truth value *false* . ¬p is also written as: neg p in ASCII.
  - implicitly, for every atom p: opposes(p, ¬p);
- Priorities between rules: partially-ordered.
  - Represent priorities via reserved predicate that compares rule labels:
    - overrides(rule1, rule2) means rule1 is higher-priority than rule2.
    - Each rule optionally has a rule label whose form is a functional term.
    - overrides can be reasoned about, just like any other predicate.

PROJECT HALO

VULCAN

Priorities are available and useful

- Priority information is naturally available and useful. E.g.,
  - recency: higher priority for more recent updates.
  - specificity: higher priority for more specific cases (e.g., exceptional cases, sub-cases, inheritance).
  - authority: higher priority for more authoritative sources (e.g., legal regulations, organizational imperatives).
  - reliability: higher priority for more reliable sources (e.g., security certificates, via-delegation, assumptions, observational data).
  - closed world: lowest priority for catch-cases.

- Many practical rule systems employ priorities of some kind, often implicit. E.g.,
  - rule sequencing in Prolog and production rules.
    - Courteous LP subsumes this as a special case (totally-ordered priorities)
    - Also Courteous LP enables: merging, more flexible & principled treatment.

PROJECT HALO

VULCAN

Courteous LP: Advantages

- Facilitate updating and merging, modularity and locality in specification.
- Expressive: strong negation, exclusions, partially-ordered prioritization, reasoning to infer prioritization.
- Guarantee consistent, unique set of conclusions.
  - Exclusion is enforced. E.g., never conclude discount is both 5% and that it is 10%, nor conclude both p and ¬p.
- Scaleable & Efficient: low computational overhead beyond ordinary LPs.
  - Tractable given reasonable restrictions (VB Datalog):
    - extra cost is equivalent to increasing v to (v+2) in Ordinary LP, worst-case.
  - By contrast, more expressive prioritized rule representations (e.g., Prioritized Default Logic) add NP-hard overhead.
- Modular software engineering:
  - Transform: CLP $\rightarrow \rightarrow$ OLP. Via simple "argumentation theory" approach.
    - Add-on to variety of OLP rule systems, with modest effort.

PROJECT HALO

VULCAN

New Theory & Algorithms for Higher-Order Defaults

- Combines Courteous + Hilog, and generalizes
- New approach to defaults: "argumentation theories"
  - Meta-rules specify when rules are defeated
  - [Wan, Grosof, Kifer, *et al.* ICLP-2009]
- Extends straightforwardly to combine with other key features
  - E.g., Frame syntax, external Actions
- Significantly improves on previous Courteous approach in other ways
  - Eliminates a complex transformation
  - Much simpler to implement
    - 20-30 background rules instead of 1000's of lines of code
  - Much faster when updating the premises
  - More flexible control of edge-case behaviors
  - Much simpler to analyze theoretically

PROJECT HALO

VULCAN

LPDA approach, continued

- More Advantages
  - 1st way to generalize defeasible LP, notably Courteous, to HiLog higher-order and F-Logic frames
  - Well-developed model theory, reducible to normal LP
  - Reducibility and well behavior results
  - Unifies almost all previous defeasible LP approaches
    - Each reformulated as an argumentation theory
  - Cleaner, more flexible and extensible semantics
    - Enables smooth and powerful integration of features
  - Leverages most previous LP algorithms & optimizations

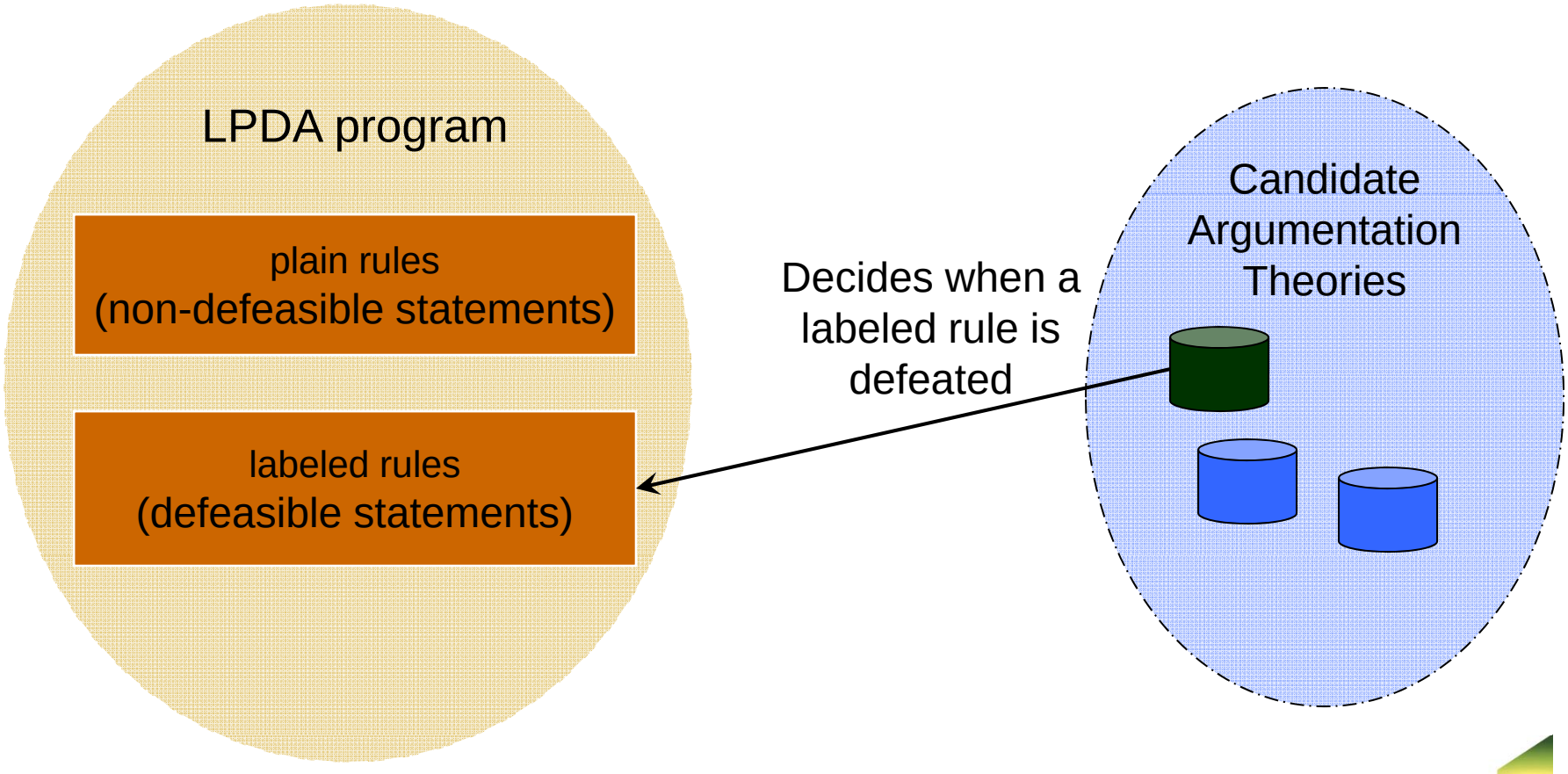
- Implemented in SILK V1 via an extension of FLORA-2
  - Public release planned for approx. winter 2009-2010

PROJECT HALO

VULCAN

LPDA Framework

- Logic Programs with Defaults and Argumentation theories



PROJECT HALO

VULCAN

Example – AT for Courteous *(AT_GCLP)*

```
$defeated(?R)     :-  $defeats(?S, ?R).
$defeats(?R, ?S) :-  $refutes(?R, ?S) or $rebuts(?R, ?S).

$refutes(?R, ?S) :-  $conflict(?R, ?S), overrides(?R, ?S).
$refuted(?R)      :-  $refutes(?R2, ?R).
$rebuts(?R, ?S)  :-  $conflict(?R, ?S),
                        not $refuted(?R), not $refuted(?S).


$candidate(?R)    :-  body(?R, ?B), call(?B).
$conflict(?R, ?S)  :-  $candidate(?R), $candidate(?S),
                           opposes(?R, ?S).
opposes(?R, ?S)   :-  opposes(?S, ?R).
opposes(?L1,?L2) :-  head(?L1, ?H), head(?L2, neg ?H).
```

PROJECT HALO

VULCAN

Outline of Talk

- Overview of SILK effort
  - Vision, Origins in parent Project Halo, Goals
  - Effort, Requirements, Approach, Roots, Status, Plans
  - Prototype, Theory, Language
- Examples and Use Cases: E-Science, E-Commerce, Trust, …
- Hyper Logic Programs KR approach and expressive features
  - More details on Requirements and Design
  - Higher-Order Defaults
    - Argumentation Theories approach (LPDA) to Defeasibility
  - Remedying FOL Semantics' Lack of Scalability
    - Weakened Classical, via Hypermonotonic mapping.
  - Comparison to other semantic rule systems and standards
    - RIF, BRMS, OWL, DBMS, etc.
- Conclusions and Directions
  - Roadmap for SILK and Industry
  - How You can be Involved

PROJECT HALO

VULCAN

Basic Hypermonotonic Mapping: clausal FOL ⇒ Courteous LP

- An FOL clause C:

  L1 or L2 or … or Lk

  is mapped to k directed clauses, one for each choice of head literal:

  L1 :- neg L2 and neg L3 and … and neg Lk

  L2 :- neg L1 and neg L3 and … and neg Lk

  …

  Lk :- neg L1 and neg L2 and … and neg Lk-1

- This is called the *omnidirectional ruleset* for C, a.k.a. the *omni*
- Conversely, a naf-free Courteous LP rule is mapped to FOL as a material implication, thus clausal. (It's fairly easy to stick to naf-free.)
- A KR S *behaves hypermonotonically* == S is nonmonotonic and when its premises are viewed classically, then entailment in S is sound but incomplete w.r.t. classical
  - Incompleteness is desirable when there's conflict

PROJECT HALO

VULCAN

Examples of Basic Hypermonotonic mapping

- /* SBVR Car rental: A driver ?p is Approved only if ?p has a Validated application. */
  - /* FOL: */ forall ?p. Validated(?p) <== Approved(?p).

  becomes the ff. omnidirectional ruleset in Hyper LP:
  - neg Approved(?p) :- neg Validated(?p) . /* Exploit strong negation feature (neg). */
  - Validated(?p) :- Approved(?p).

- /* OWL 2 DL beyond RL: The classes Cat and Bird are disjoint. */
  - /* FOL */ forall ?x. neg (Cat(?x) and Bird(?x)).

  becomes the ff. omnidirectional ruleset in Hyper LP:
  - neg Cat(?x) :- Bird(?x).
  - neg Bird(?x) :- Cat(?x).

- /* Scheduling: Joe's meeting will be at 3pm or 4pm or 5pm today. */
  - /* FOL source: */ mtg(3p) or mtg(4p) or mtg(5p).

  becomes the ff. omnidirectional ruleset in Hyper LP:
  - mtg(5p) :- neg mtg(3p) and neg mtg(4p).
  - mtg(4p) :- neg mtg(3p) and neg mtg(5p).
  - mtg(3p) :- neg mtg(4p) and neg mtg(5p).

PROJECT HALO

VULCAN

Preview: Hypermon. Mapping from full FOL

- Greatly generalizes the approach of Description LP and OWL 2 RL
- Leverages generalized higher-order defaults feature of Hyper LP
- Each FOL clause is mapped to a small set of LP rules (defaults)

- Covers FOL unrestricted clauses (not just Horn)
- Can further add skolemization, thus cover full FOL
- Can further add Higher-order and Frames, thus cover "FOL++"

- Thus can cover full OWL/RDF, full Common Logic, most of SBVR

- Give up disjunction / reasoning by cases, so is weakened

- Hyper LP handles conflict robustly

PROJECT HALO

VULCAN

Remedying FOL Semantics' Lack of Scalability

- Hyper LP handles conflict robustly
  - Whereas FOL is a "Glass Bubble" – it's perfectly brittle semantically in face of contradictions from …
  - Quality problems/errors in the data and knowledge
  - Conflict when merging KBs

    E.g., OWL beyond the RL subset suffers this problem

A VLKB with a million or billion axioms formed by merging from multiple Web sources, is unlikely to have zero KB/KA conflicts from:

- Human knowledge entry/editing
- Implicit context, cross-source ontology interpretation
- Updating cross-source
- Source trustworthiness

- *Weakening provides a critical advantage for VLKB scalability*
  - *semantically, as well as computationally*

PROJECT HALO

VULCAN

FOL: A Glass Bubble

Extreme sensitivity to conflict limits its scalability in # of axioms and # of merges

Keijo Kopra from Finland as he competes in the Iittala Cup glass-blowing competition June 7, 2008. (Reuters)

KR Conflict Handling – A Key to Scalability

BEFORE		AFTER
KR: Classical Logic (FOL, OWL)	⇒	KR: LP with Defaults (Courteous-style)

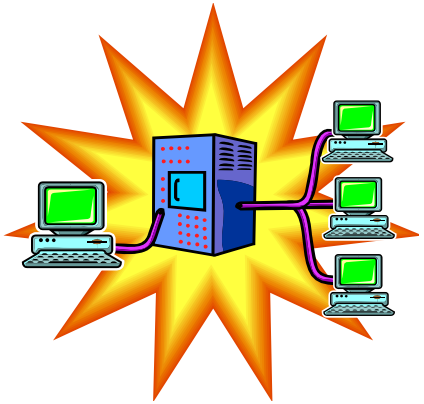
Contradictory conflict is globally contagious, invalidates all results.

⇒

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

Knowledge integration involving conflict is labor-intensive, slow, costly.

⇒



Knowledge integration involving conflict is highly automated, faster, cheaper.

Outline of Talk

- Overview of SILK effort
  - Vision, Origins in parent Project Halo, Goals
  - Effort, Requirements, Approach, Roots, Status, Plans
  - Prototype, Theory, Language
- Examples and Use Cases: E-Science, E-Commerce, Trust, …
- Hyper Logic Programs KR approach and expressive features
  - More details on Requirements and Design
  - Higher-Order Defaults
    - Argumentation Theories approach (LPDA) to Defeasibility
  - Remedying FOL Semantics' Lack of Scalability
    - Weakened Classical, via Hypermonotonic mapping.
  - Comparison to other semantic rule systems and standards
    - RIF, BRMS, OWL, DBMS, etc.
- Conclusions and Directions
  - Roadmap for SILK and Industry
  - How You can be Involved

PROJECT HALO

VULCAN

Providing Declarative Semantics for Procedural Attachments

- Procedural attachments historically viewed in KR theory as … well … *procedural* ;-) … rather than declarative.
  - Not much theoretical attention
- Needed for Semantic Web: a *declarative* KR approach to them
- Production LP is probably the most important approach today
  - E.g., SILK, RuleML, SweetRules, IBM Common Rules, predecessors
    - Formerly called ***Situated*** LP
  - Provides disciplined expressive abstraction for two broad, often-used categories of procedural attachments:
    - External Queries: Purely-informational – permitted in rule bodies
    - External Actions : Side-effectful – permitted in rule heads
  - Makes restrictions: assumptions become explicit
  - Declarative semantic guarantees, interoperability
  - Embodies primarily analytical insight, initially
  - Provides also: expressive generalizations, algorithms/techniques

PROJECT HALO

VULCAN

Updating & Events in Production LP

- "Event" is a set of facts/rules, constituting an update to KB
- An interesting kind of thing to do with a Production LP is to update its premises, and perform incremental inferencing+action.
  - new PLP *P2* = (update *U2*) ∪ (previous *P1*)
  - Incremental inferencing+action is defined as:
    - Generate the inferences that are novel

      NovelConclusions = Conclusions(*P2*) – Conclusions(*P1*)
    - Perform the external actions (effecting) associated with NovelConclusions
- Extension to PLP:
  - An event channel is an attached procedure that delivers events as updates
    - Listening to an event channel can be viewed as a persistent external query

PROJECT HALO

VULCAN

Outline of Talk

- Overview of SILK effort
  - Vision, Origins in parent Project Halo, Goals
  - Effort, Requirements, Approach, Roots, Status, Plans
  - Prototype, Theory, Language
- Examples and Use Cases: E-Science, E-Commerce, Trust, …
- Hyper Logic Programs KR approach and expressive features
  - More details on Requirements and Design
  - Higher-Order Defaults
    - Argumentation Theories approach (LPDA) to Defeasibility
  - Remedying FOL Semantics' Lack of Scalability
    - Weakened Classical, via Hypermonotonic mapping.
  - Comparison to other semantic rule systems and standards
    - RIF, BRMS, OWL, DBMS, etc.
- Conclusions and Directions
  - Roadmap for SILK and Industry
  - How You can be Involved

PROJECT HALO

VULCAN

Semantic Rules KR: Features Comparison

Level ("generation")	Groups of features	*SILK V1*	*Flora*	*RIF-BLD*
1G. Basic	ie: Horn, chaining, external queries, built-ins *(Level Summary)*	Y	Y	Y
2G. Advanced	*(Level Summary)*	Most!	lots	some
	Equality (derived via non-fact rules)	Y	Y	Y
	Functions	Y	Y	Y
	Convenience Package: Frames, integrity constraints, skolemization	Y	Y	R. frames
	Closed-World: unstratified NAF, aggregates, Lloyd-Topor	Y	Y	N
	Higher-Order (incl. reification)	Y	Y	N
	Actions (external) (via procedural attachments)	Developing	N	N
	Base Defaults (prioritized, cf. Courteous)	Y	N	N
	Webized syntax (URI names and XML/RDF KBs)	Developing	N	Y
3G. Hyper	*(Level Summary)*	Pioneer!	N	N
	Higher-Order Defaults	Y	N	N
	Weakened Classical (sound interchange with default rules)	Developing	N	N
Other Misc.		(NA)	(NA)	(NA)
	Other Expressive	Developing	R. inherit.	-
	Reasoner Efficiency (upper-tier on OpenRuleBench)	good	good	NA (standard)

- Summarizes detailed analysis of 40 KR expressive features, 17 systems.
- Notes: R. = Restricted; RIF-BLD = W3C Rule Interchange Format - Basic Logic Dialect.

57

Features Comparison – More Systems & Stds

Level	Groups of Features	*SILK1*	*Flora*	*RIF-BLD*	*Jena*	*Onto-broker*	*Jess*	*IBM C.R.*	*DLV*	*SQL*	*SPA-RQL*	*Common Logic*	*OWL2 RL*	*OWL2 DL*
Basic	Horn chain. etc.	Y	Y	Y	Y	Y	Y	Y	Y	R.	R.	Y	R.	R.
Advanced	*(Level summary)*	Most!	lots	some	some	some	some	some	some	some	some	some	some	some
	Equality	Y	Y	Y	R.	R.	R.	N	Y	R.	R.	Y	R.	Y
	Functions	Y	Y	Y	N	N	N	Y	Y	N	N	Y	N	N
	Frames etc.	Y	Y	R.	R.	R.	R.	R.	R.	R.	R.	R.	R.	R.
	Closed-World	Y	Y	N	N	Y	R.	R.	most	R.	R.	N	N	N
	Higher-Order	Y	Y	N	N	N	R.	N	N	R.	R.	Y	R. bit	R. bit
	Actions	Dev.	N	N	N	N	Y	Y	N	N	N	N	N	N
	Base Defaults	Y	N	N	N	N	N	Y	N	N	N	N	N	N
	Webized	Dev.	R.	Y	Y	R.	R.	R.	R.	N	Y	Y	Y	Y
Hyper	*(Level summary)*	1st!	N	N	N	N	N	N	N	N	N	N	N	N
	H-O. Defaults	Y	N	N	N	N	N	N	N	N	N	N	N	N
	Weak. Classi.	Dev.	N	N	N	N	N	N	N	N	N	N	N	N
Misc.		NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
	Other Expres.	Dev.	inherit.	-	-	-	events	-	disju.	R.	R.	classical	-	classic.
	Efficiency	good	good	NA	fair	good	fair	poor	good	NA	NA	NA	NA	NA

- Summarizes detailed analysis of 40 KR expressive features, 17 systems.
- Notes: Dev. = Developing, R. = Restricted; C.R.=Common Rules; disju.=disjunctive.

58

Features Comparison – More Systems & Stds

Level	Groups of Features	SILK1	Flora	RIF-BLD	Jena	Onto-broker	Jess	IBM C.R.	DLV	SQL	SPA-RQL	Common Logic	OWL2 RL	OWL2 DL
Basic	Horn chain. etc.													
Advanced	*(Level summary)*													
	Equality													
	Functions													
	Frames etc.													
	Closed-World													
	Higher-Order													
	Actions													
	Base Defaults													
	Webized													
Hyper	*(Level summary)*													
	H-O. Defaults													
	Weak. Classi.													
Misc.														
	Other Expres.													
	Efficiency													

Background on Systems and Standards:

- Jess is a representative commercial production rule (PR) system. PR was shown 5-7 years ago to have a semantic subset (based on the SweetRules translation). The currently most commercially important business rule management systems (BRMS) are based on PR or similar event-condition (ECA) action rules.

- W3C Rule Interchange Format (RIF)'s Basic Logic Dialect (BLD) is its main semantic part. There is also a framework for extensions. RIF is based primarily on RuleML, except for RIF's Production Rule Dialect (PRD).

- W3C OWL 2 RL is OWL's Rules subset (based on Description LP).

- Jena is a popular open-source semantic web toolkit, incl. for rules.

- Ontobroker is a commercial forward-chaining LP system.

- IBM Common Rules (C.R.) introduced the base defaults feature.

- Common Logic (CL) is an ISO standard for classical logic, used also by OMG's Semantic Business Vocabulary and Rules (SBVR) standard.

- DLV is a disjunctive LP system, by Univ. of Calabria (it has OR in rule heads)

Features Comparison – More Systems & Stds

Level	Groups of Features	SILK1	Flora	RIF-BLD	Jena	Onto-broker	Jess	IBM C.R.	DLV	SQL	SPA-RQL	Common Logic	OWL2 RL	OWL2 DL
Basic	Horn chain. etc.	Y	Y	Y	Y	Y	Y	Y	Y	R.	R.	Y	R.	R.
Advanced	*(Level summary)*	Most!	lots	some	some	some	some	some	some	some	some	some	some	some
	Equality	Y	Y	Y	R.	R.	R.	N	Y	R.	R.	Y	R.	Y
	Functions	Y	Y	Y	N	N	N	Y	Y	N	N	Y	N	N
	Frames etc.	Y	Y	R.	R.	R.	R.	R.	R.	R.	R.	R.	R.	R.
	Closed-World	Y	Y	N	N	Y	R.	R.	most	R.	R.	N	N	N
	Higher-Order	Y	Y	N	N	N	R.	N	N	R.	R.	Y	R. bit	R. bit
	Actions	Dev.	N	N	N	N	Y	Y	N	N	N	N	N	N
	Base Defaults	Y	N	N	N	N	N	Y	N	N	N	N	N	N
	Webized	Dev.	R.	Y	Y	R.	R.	R.	R.	N	Y	Y	Y	Y
Hyper	*(Level summary)*	1st!	N	N	N	N	N	N	N	N	N	N	N	N
	H-O. Defaults	Y	N	N	N	N	N	N	N	N	N	N	N	N
	Weak. Classi.	Dev.	N	N										
Misc.		NA	NA	NA										
	Other Expres.	Dev.	inherit.	-	-	-	events	-	disju.	R.	R.	classical	-	classic.
	Efficiency	good	good	NA	fair	good	fair	poor	good	NA	NA	NA	NA	NA

More features than any other

- Summarizes detailed analysis of 40 KR expressive features, 17 systems.
- Notes: Dev. = Developing, R. = Restricted; C.R.=Common Rules; disju.=disjunctive.

PROJECT HALO

VULCAN

60

Features Comparison – More Systems & Stds

Level	Groups of Features	SILK1	Flora	RIF-BLD	Jena	Onto-broker	Jess	IBM C.R.	DLV	SQL	SPA-RQL	Common Logic	OWL2 RL	OWL2 DL
Basic	Horn chain. etc.	Y	Y	Y	Y	Y	Y	Y	Y	R.	R.	Y	R.	R.
Advanced	*(Level summary)*	Most!	lots	some	some	some	some	some	some	some	some	some	some	some
	Equality	Y	Y	Y	R.	R.	R.	N	Y	R.	R.	Y	R.	Y
	Functions	Y	Y	Y	N	N	N	Y	Y	N	N	Y	N	N
	Frames etc.	Y	Y	R.	R.	R.	R.	R.	R.	R.	R.	R.	R.	R.
	Closed-World	Y	Y	N	N	Y	R.	R.	most	R.	R.	N	N	N
	Higher-Order	Y	Y	N	N	N	R.	N						
	Actions	Dev.	N	N	N	N	Y	Y						
	Base Defaults	Y	N	N	N	N	N	Y						
	Webized	Dev.	R.	Y	Y	R.	R.	R.						
Hyper	*(Level summary)*	1st!	N	N	N	N	N	N						
	H-O. Defaults	Y	N	N	N	N	N	N						
	Weak. Classi.	Dev.	N	N	N	N	N	N	N	N	N	N	N	N
Misc.		NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
	Other Expres.	Dev.	inherit.	-	-	-	events	-	disju.	R.	R.	classical	-	classic.
	Efficiency	good	good	NA	fair	good	fair	poor	good	NA	NA	NA	NA	NA

Much more expressive than production/ECA rules

- Summarizes detailed analysis of 40 KR expressive features, 17 systems.
- Notes: Dev. = Developing, R. = Restricted; C.R.=Common Rules; disju.=disjunctive.

PROJECT HALO

VULCAN

61

Features Comparison – More Systems & Stds

Level	Groups of Features	SILK1	Flora	RIF-BLD	Jena	Onto-broker	Jess	IBM C.R.	DLV	SQL	SPA-RQL	Common Logic	OWL2 RL	OWL2 DL
Basic	Horn chain. etc.	Y	Y	Y	Y	Y	Y	Y	Y	R.	R.	Y	R.	R.
Advanced	*(Level summary)*	Most!	lots	some	some	some	some	some	some	some	some	some	some	some
	Equality	Y	Y	Y	R.	R.	R.	N	Y	R.	R.	Y	R.	Y
	Functions	Y	Y	Y	N	N	N	Y	Y	N	N	Y	N	N
	Frames etc.	Y	Y	R.	R.	R.	R.	R.	R.	R.	R.	R.	R.	R.
	Closed-World	Y	Y	N	N	Y	R.	R.	most	R.	R.	N	N	N
	Higher-Order	Y	Y	N	N	N	R.	N	N	R.	R.	Y	R. bit	R. bit
	Actions	Dev.	N	N	N	N	Y	Y	N	N	N	N	N	N
	Base Defaults	Y	N	N	N	N	N	Y	N	N	N	N	N	N
	Webized	Dev.	R.	Y	Y	R.	R.	R.	R.	N	Y	Y	Y	Y
Hyper	*(Level summary)*													
	H-C													
	Wea													
Misc.														
	Oth													
	Effi													

NEWLY COMBINES previous advanced features:
e.g., {full Frames + Base Defaults}
+ {full Closed-World + Actions}
+ {fully Webized + good Efficiency}

- Notes: Dev. = Developing, R. = Restricted; C.R.=Common Rules; disju.=disjunctive.

Features Comparison – More Systems & Stds

Level	Groups of Features	*SILK1*	*Flora*	*RIF-BLD*	*Jena*	*Onto-broker*	*Jess*	*IBM C.R.*	*DLV*	*SQL*	*SPA-RQL*	*Common Logic*	*OWL2 RL*	*OWL2 DL*
Basic	Horn chain. etc.	Y	Y	Y	Y	Y	Y	Y	Y	R.	R.	Y	R.	R.
Advanced	*(Level summary)*	Most!	lots	some	some	some	some	some	some	some	some	some	some	some
	Equality	Y	Y	Y	R.	R.	R.	N	Y	R.	R.	Y	R.	Y
	Functions	Y	Y	Y	N	N	N	Y	Y	N	N	Y	N	N
	Frames etc.	Y	Y	R.	R.	R.	R.	R.	R.	R.	R.	R.	R.	R.
	Closed-World	Y	Y	N	N	Y	R.	R.	most	R.	R.	N	N	N
	Higher-Order	Y	Y	N	N	N	R.	N	N	R.	R.	Y	R. bit	R. bit
	Actions	Dev.	N	N	N	N	Y	Y	N	N	N	N	N	N
	Base Defaults	Y	N	N	N	N	N	Y	N	N	N	N	N	N
	Webized	Dev.	R.	Y	Y	R.	R.	R.	R.	N	Y	Y	Y	Y
Hyper	*(Level summary)*	1st!	N											
	H-O. Defaults	Y	N											
	Weak. Classi.	Dev.	N											
Misc.		NA	NA											
	Other Expres.	Dev.	inh											
	Efficiency	good	good	NA	fair	good	fair	poor	good	NA	NA	NA	NA	NA

Advanced-Level DELTAS w.r.t. Flora:
V1: Base Defaults;
V2 (in Dev.): Actions, Webized

- Summarizes detailed analysis of 40 KR expressive features, 17 systems.
- Notes: Dev. = Developing, R. = Restricted; C.R.=Common Rules; disju.=disjunctive.

63

Features Comparison – More Systems & Stds

Level	Groups of Features	*SILK1*	*Flora*	*RIF-BLD*	*Jena*	*Onto-broker*	*Jess*	*IBM C.R.*	*DLV*	*SQL*	*SPA-RQL*	*Common Logic*	*OWL2 RL*	*OWL2 DL*
Basic	Horn chain. etc.	Y	Y	Y	Y	Y	Y	Y	Y	R.	R.	Y	R.	R.
Advanced	*(Level summary)*	Most!	lots	some	some	some	some	some	some	some	some	some	some	some
	Equality	Y	Y	Y	R.	R.	R.	N	Y	R.	R.	Y	R.	Y
	Functions	Y	Y	Y										
	Frames etc.	Y	Y	R										
	Closed-World	Y	Y	N										
	Higher-Order	Y	Y	N										
	Actions	Dev.	N	N										
	Base Defaults	Y	N	N										
	Webized	Dev.	R.	Y	Y	R.	R.	R.	R.	N	Y	Y	Y	Y
Hyper	*(Level summary)*	1st!	N	N	N	N	N	N	N	N	N	N	N	N
	H-O. Defaults	Y	N	N	N	N	N	N	N	N	N	N	N	N
	Weak. Classi.	Dev.	N	N	N	N	N	N	N	N	N	N	N	N
Misc.		NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
	Other Expres.	Dev.	inherit.	-	-	-	events	-	disju.	R.	R.	classical	-	classic.
	Efficiency	good	good	NA	fair	good	fair	poor	good	NA	NA	NA	NA	NA

FUNDAMENTALLY NEW
Hyper-Level features:
V1: Higher-Order Defaults;
V2 (in Dev.): Weakened Classical

- Summarizes detailed analysis of 40 KR expressive features, 17 systems.
- Notes: Dev. = Developing, R. = Restricted; C.R.=Common Rules; disju.=disjunctive.

64

KR Features Comparison: Cyc

- SILK also draws upon Cyc
  - Plenty to learn from Cyc's design and experience
- Cyc lacks (as yet) a well-understood semantics, so it's not quite a semantic rule system
  - Previously, Cycorp has described it both in terms of FOL and defaults
  - However, preliminary indications from the ReCyc translation effort indicates Cyc's KR is closer in spirit to LP than to Classical
- Cyc's set of KR features correspond roughly to SILK's
  - This provides some confirmation for SILK's goals w.r.t. features

Outline of Talk

- Overview of SILK effort
  - Vision, Origins in parent Project Halo, Goals
  - Effort, Requirements, Approach, Roots, Status, Plans
  - Prototype, Theory, Language
- Examples and Use Cases: E-Science, E-Commerce, Trust, …
- Hyper Logic Programs KR approach and expressive features
  - More details on Requirements and Design
  - Higher-Order Defaults
    - Argumentation Theories approach (LPDA) to Defeasibility
  - Remedying FOL Semantics' Lack of Scalability
    - Weakened Classical, via Hypermonotonic mapping.
  - Comparison to other semantic rule systems and standards
    - RIF, BRMS, OWL, DBMS, etc.
- Conclusions and Directions
  - Roadmap for SILK and Industry
  - How You can be Involved

PROJECT HALO

VULCAN

BRMS Industry Roadmap: facing disruption

- Semantic rules is a prospectively truly disruptive innovation for the existing business rules management systems (BRMS) industry sector
- See "The New Rules of Business" [Grosof EBRC-2007 keynote]
  - Strategic analysis of evolving market dynamics and what players should do about it
    - Done with a Management professor hat on
  - http://www.mit.edu/~bgrosof/#EBRC2007Talk

PROJECT HALO

VULCAN

Impact Opportunities for SILK and HalAR

- Improve by orders of magnitude:
  - Scale of practical semantic default+actions reasoning
    - <~1000 rules ⇒ ?100,000+ rules
  - Collaboration costs of multifold KB merging when there's conflict (as is usual)
    - Can take human out of the loop at run time
  - Population of users capable of specifying semantic rules
    - "KR Power to the People!" Leverage Aura and SMW+ KA/UI front-ends.
- Synergize best of last 20 years of pure-research progress in LP KR
  - ⇒ Redefine KR playing field of semantic web, business rules, & process management
- Provide a key missing research piece for SOA / web services
  - Enable building shared business/govt KBs on processes & policies ⇒ virtuous circle
- Hope: be like advance of the Relational model in DBMS
  - Will Hyper LP be to the 2010s what Relational was to 1970s-80s?

Key KR infrastruct. for widely-authored VLKBs for science and business that answer questions, proactively supply information, and reason powerfully

PROJECT HALO

VULCAN

Future Directions for SILK

- Process – more complex
- Explanation and Abduction – for science and UI
- Truth maintenance, Termination
- Natural Language KA and UI
  - Controlled English
- Uncertainty, Disjunction, Constraints
- Distributed and Parallel reasoning
- More Connectors & Translations to other KR/DB/services
- Meta-reasoning for control of inferencing
- And Use Cases, of course
- Halo is part of an increasingly-integrated strategy at Vulcan to invest in semantics and advanced knowledge tools
  - Other investments: Radar Networks, ZoomInfo, Evri, etc.
  - Semantic MediaWiki+ is an early spinout

PROJECT HALO

VULCAN

How You can be Involved

- General Contact: Benjamin Grosof benjaming@vulcan.com
  - Suggest design, use cases, experts, cooperations
- Visit the SILK webpage and sign up for the mailing list so you'll be alerted of announcements about SILK
  - URL: http://silk.semwebcentral.org
  - Mailing list: silk-announce@semwebcentral.org (very low volume)
- Provide comments on SILK language design
  - Initial public draft in early 2010; selected earlier reviewers starting Nov. 2009
  - Plan to propose a RIF extension with defaults and actions
    - Corresponding to a large expressive subset of SILK
- Try out SILK software
  - Prototype, free for research use
  - V2 release in 2010; selected earlier users of the alpha starting Nov. 2009
  - Upcoming: refereed demos of SILK at ISWC-2009 and RuleML-2009

PROJECT HALO

VULCAN

Acknowledgements

- SILK contributors
  - (previously listed)

- Contributors to several key previous KR efforts
  - RuleML and SWSL (Semantic Web Services Language) standards designs
  - SweetRules and Flora-2 systems

- Especially:
  - Michael Kifer, Mark Greaves, Dave Gunning, Mike Dean, Hui Wan, Paul Haley, Vinay Chaudhri, David Martin, Keith Goolsbey, Harold Boley, Said Tabet, and Guizhen Yang

PROJECT HALO

VULCAN

SILK – Transforming Knowledge

Thank You

Disclaimer: The preceding slides represent the views of the author only.
All brands, logos and products are trademarks or registered trademarks of their respective companies.

PROJECT HALO

VULCAN