Version of June 18, 2009; may contain small edits beyond the version in the conference materials

SILK: Semantic Rules Take the Next Big Step in Power

Benjamin Grosof*

June 18, 2009

1-Hour presentation at SemTech 2009**

* Vulcan Inc., benjaming@vulcan.com

http://silk.projects.semwebcentral.org

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

** 2009 Semantic Technology Conference, San Jose, CA,
http://www.semantic-conference.com

PROJECT HALO

VULCAN

Outline of Talk

- Overview
  - Vision, Origins, Goals, Effort, Approach, Roots, Status
  - V1 Prototype, Theory, Language; V2 plans
  - Examples and Use Cases

- Drill down on the KR Language and System
  - Requirements analysis
  - Hyper Logic Programs KR approach and expressive features
    - Higher-Order Defaults. 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

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” – The Name

- “Semantic Inferencing on Large Knowledge”
- What the next generation Web will be spun from

PROJECT HALO

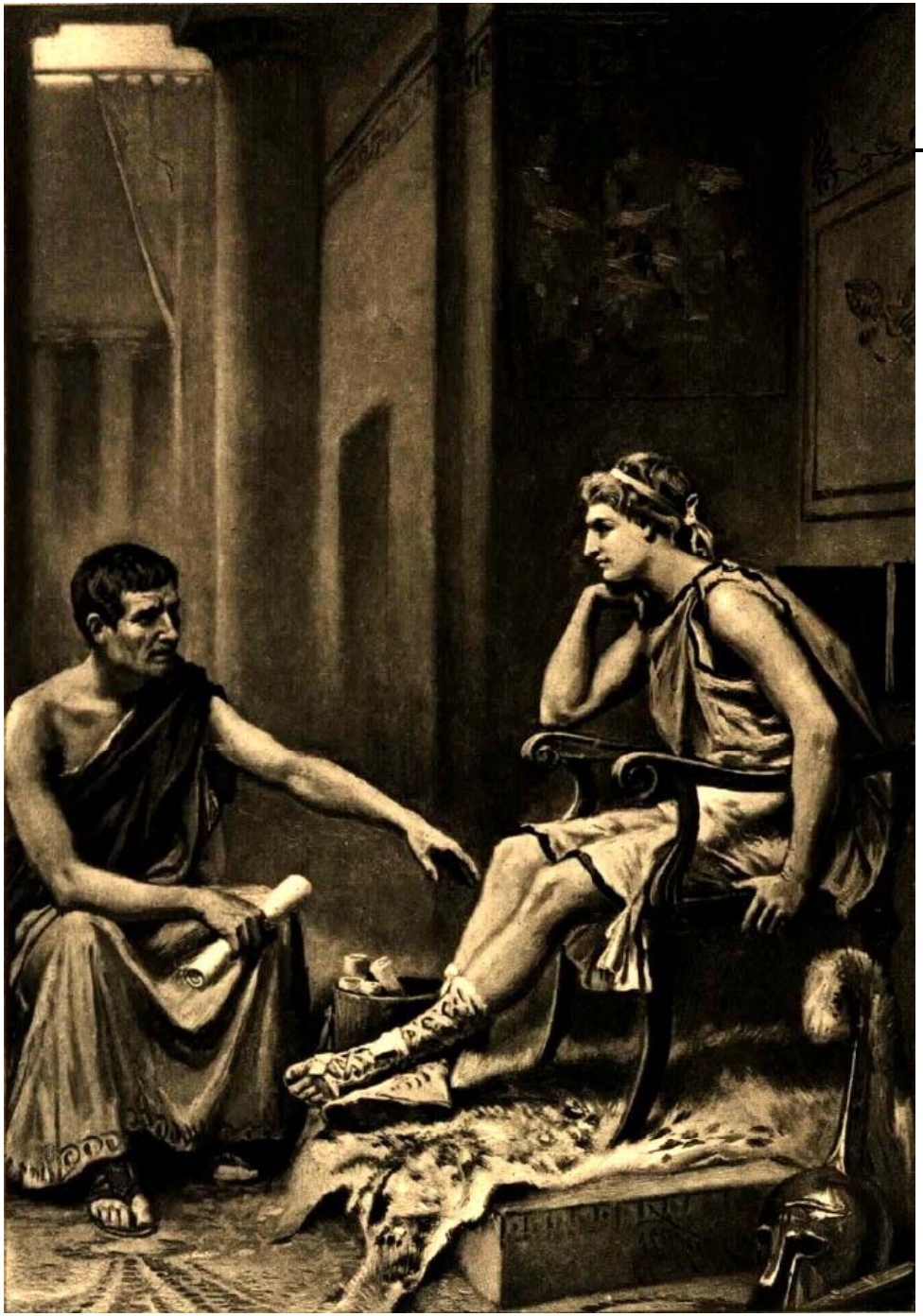
VULCAN

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 Effort Overview

- Begun in 2008
  - Part of Halo Advanced Research (HalAR), the new half of Project Halo
- Largest rule research program in the US (that we're aware of)
  - Primarily via contractors
- Structured Knowledge as initial focus
- KR System with multiple software components
  - Logical Language, incl. Syntax and Semantics
  - Reasoning, incl. Backward and Forward Inferencing
  - Web Knowledge Interchange, incl. Translators
  - KA/UI Support, incl. for Editing and Explanation
- Evolutionary Approach
  - Start from known core KR
  - Add more features in principled fashion
  - Requirements, use cases, benchmarking, KB building;
    system design (incl. theory, usability), implementation, testing (incl. task)

PROJECT HALO

VULCAN

SILK Language Starting Point is LP

- Declarative Logic Programs (LP) is starting point for SILK language
  - Normal LP, with well-founded semantics. A rule has the form:

    H :- B_1 and … and B_k and not* B_{k+1} and … and not B_m . (H, B_i are atoms**)

* "not" here means closed-world negation, i.e., "negation as failure (naf)", a.k.a. "weak" negation

** An atom has the form: predicate(argument_term_1, …, argument_term_N)

e.g., height(Joe, multiply(170, centimeter))

PROJECT HALO

VULCAN

LP is the Core KR in today's world … incl. Sem. Web

- LP is the core logical 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 are based on semantic rules: Description Logic Programs (DLP) + moderate extensions. Oracle, for example.
    - OWL 2** standard 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 probably just didn't realize it!

** W3C Last Call Working Draft

Why the Sem. Tech. Industry Needs something like SILK

- Need to raise abstraction level, e.g., for SME and NL KA/UI
- Need robustness & meta-reasoning for web KB integration
  - Cope with conflict, mediation, context, knowledge quality
  - Defaults $\Rightarrow$ robustness, modularity $\Rightarrow$ scalability
  - Higher-order $\Rightarrow$ 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

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)
- Ontoprise GmbH (Raphael Volz, Jurgen Angele, Daniel Hansch)
- Automata (Paul Haley)
- Cycorp (Keith Goolsbey, Doug Lenat, Ben Rode)
- Boeing (Peter Clark)
- University of Texas (Bruce Porter)
- University of Toronto (Sheila McIlraith; students H. Ghaderi, S. Sohrabi)
- University of Amsterdam (Bert Bredeweg)
- University of Freiburg (Georg Lausen)
- University of Michigan (Michael Wellman)
- Richard Fikes, consultant (Stanford University)
- (More to come in 2009)

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

SILK V1: Overview

- Completed in fall 2008, and refined since
- Implementation: Prototype Hyper LP rule engine
  - Extends Flora-2 system to add higher-order defaults
    - Flora-2, from Stony Brook Univ., included a strong set of advanced features as a point of departure. It's written on top of XSB, a mature Prolog written in C.
- Language specification (partial)
  - Covers most of the major expressive features
  - Semantics for Higher-Order Defaults, and several other novel feature combinations
- Theory and algorithms for Higher-order Defaults
  - The most fundamental new aspect of Hyper LP

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

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. Near term steps include:
  - Add expressive features, e.g., Weakened Classical, external Actions
  - Webize more fully, e.g., knowledge interchange, UI

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).
  !- fishCount(?s,?r,?f,?C1) and 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  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. !- prefixes an exclusion, and means "it's a conflict if". In an exclusion, | means given that.

E-Commerce Ex. of Causal Process Reas. in SILK

```
/*    E-commerce delivery logistics. */
/* Initial fact, and prevention constraint that location is unique */
  loc(s0,PlasmaTV46,LasVegasWH).
  !- loc(?s,?item,?posn1) and 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 */
  @pel1  loc(?s+1,?item,?posn) :- loc(?s,?item,?posn).
  @pel2  neg loc(?s+1,?item,?posn) :- neg loc(?s,?item,?posn).
/* Action effect axiom has higher priority than the persistence axioms */
  overrides(mov1,pel1).   overrides(mov1,pel2).
/* An action instance occurs */
  @deliv57  shipment(s0+1,PlasmaTV46, WH_LasVegasNV, 9_Fog_St_SeattleWA).
```

As desired: |= loc(s0+2,PlasmaTV46, 9_Fog_St_SeattleWA) and
neg loc (s0+2,PlasmaTV46, WH_LasVegasNV).

Notes: @ prefixes a rule label. ? prefixes a variable. :- means if. !- prefixes an exclusion, and means "it's a conflict if". In an exclusion, | means given that.

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). webPage(Al).

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

Notes: @ prefixes a rule label. ? prefixes a variable. :- means if. !- prefixes an exclusion, and means "it's a conflict if". In an exclusion, | means given that.

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
  - Vision, Origins, Goals, Effort, Approach, Roots, Status
  - V1 Prototype, Theory, Language; V2 plans
  - Examples and Use Cases

- Drill down on the KR Language and System
  - Requirements analysis
  - Hyper Logic Programs KR approach and expressive features
    - Higher-Order Defaults. 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

More Rationale about LP as Starting Point KR

- 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
- Can realistically hope to be web-scalable performance-wise, unlike highly expressive classical
  - Polynomial computational complexity, under non-onerous restrictions
  - Many optimizations available
  - Established track record of high scalability for relational databases

PROJECT HALO

VULCAN

What One Gives Up by choosing LP as Starting Point

- "Disjunction", i.e., Reasoning By Cases
  - By contrast:
    - LP concludes (A or B) only if conclude A or conclude B.
    - LP prohibits disjunction in head of rule.
- Disjunction is a source of exponential computational complexity (worst-case), when unrestricted
  - Classical logic is NP-complete, even for propositional (3-SAT)
  - Major disjunctive LP approaches are, too
  - Stable semantics for LP is, too (for unstratified, when it diverges from well founded)
- Can hope to reintroduce disjunction in restricted or altered form, or develop work-arounds
- But there are many apps not requiring it, e.g., DBMS, BRMS

PROJECT HALO

VULCAN

Major SILK Requirements on 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

Major SILK Requirements on Expressiveness (cont.'d)

- 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

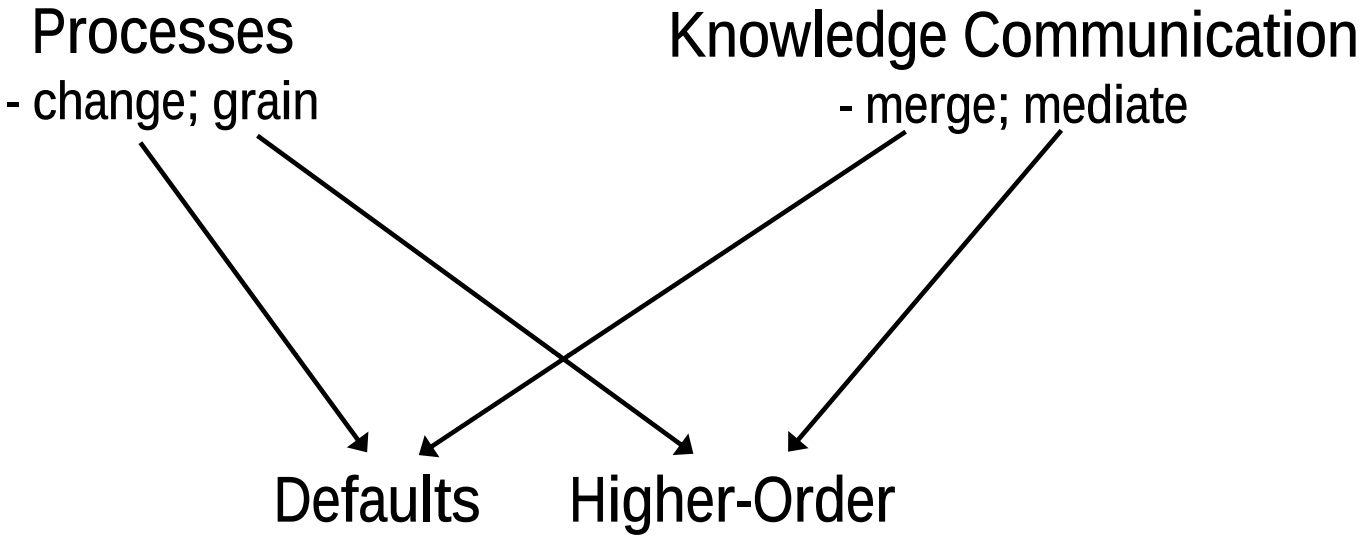
SILK Other Reasoning Requirements

- 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

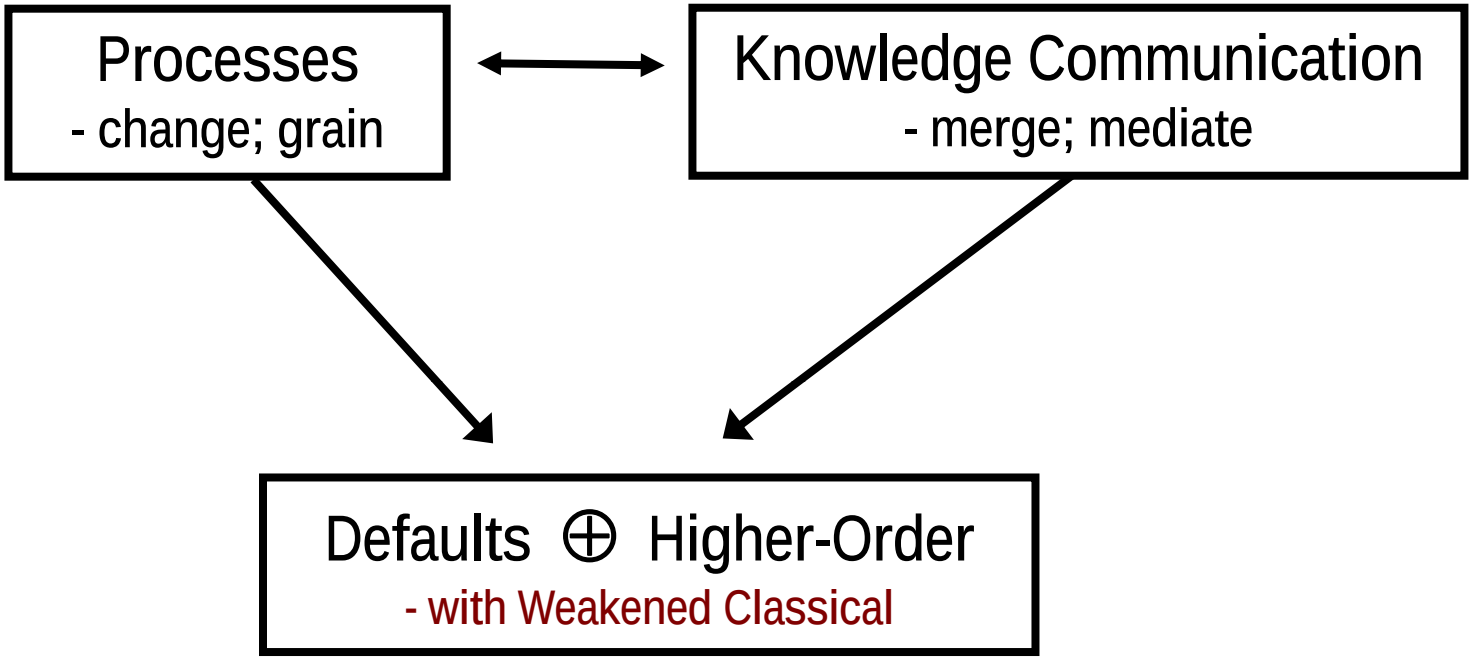


PROJECT HALO

VULCAN

32

Dependencies among Requirements II

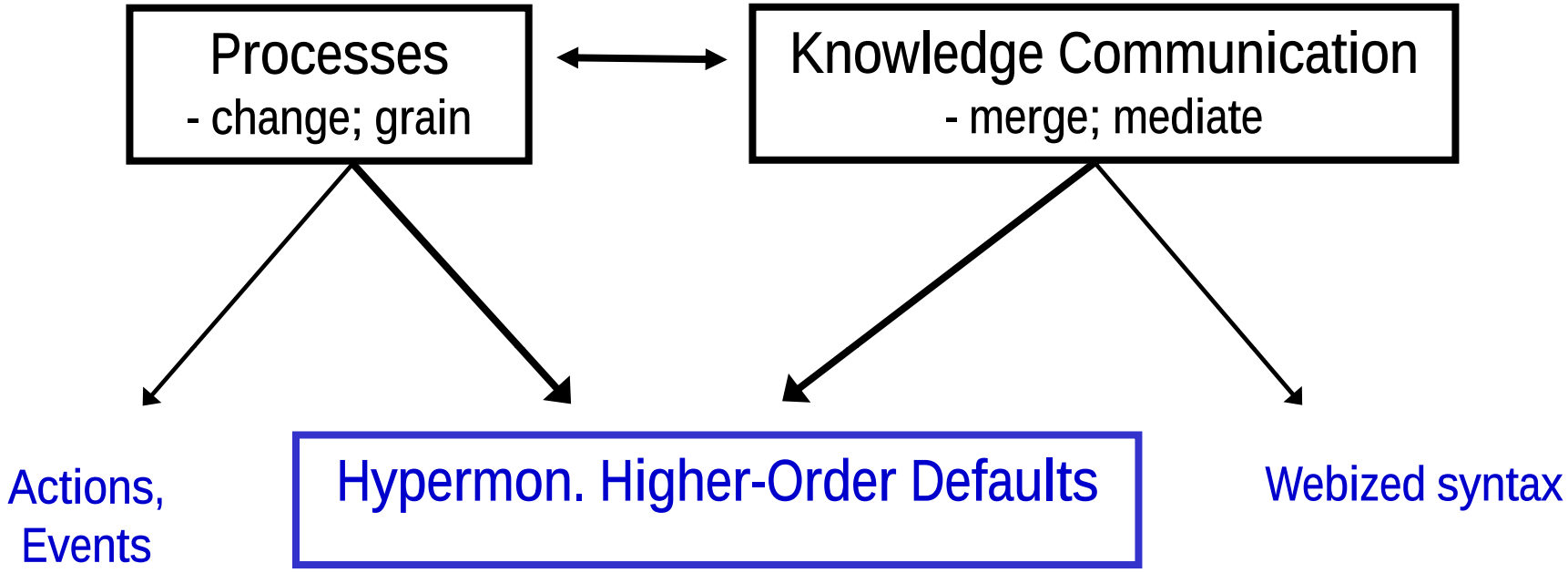


PROJECT HALO

VULCAN

33

Dependencies among Requirements III

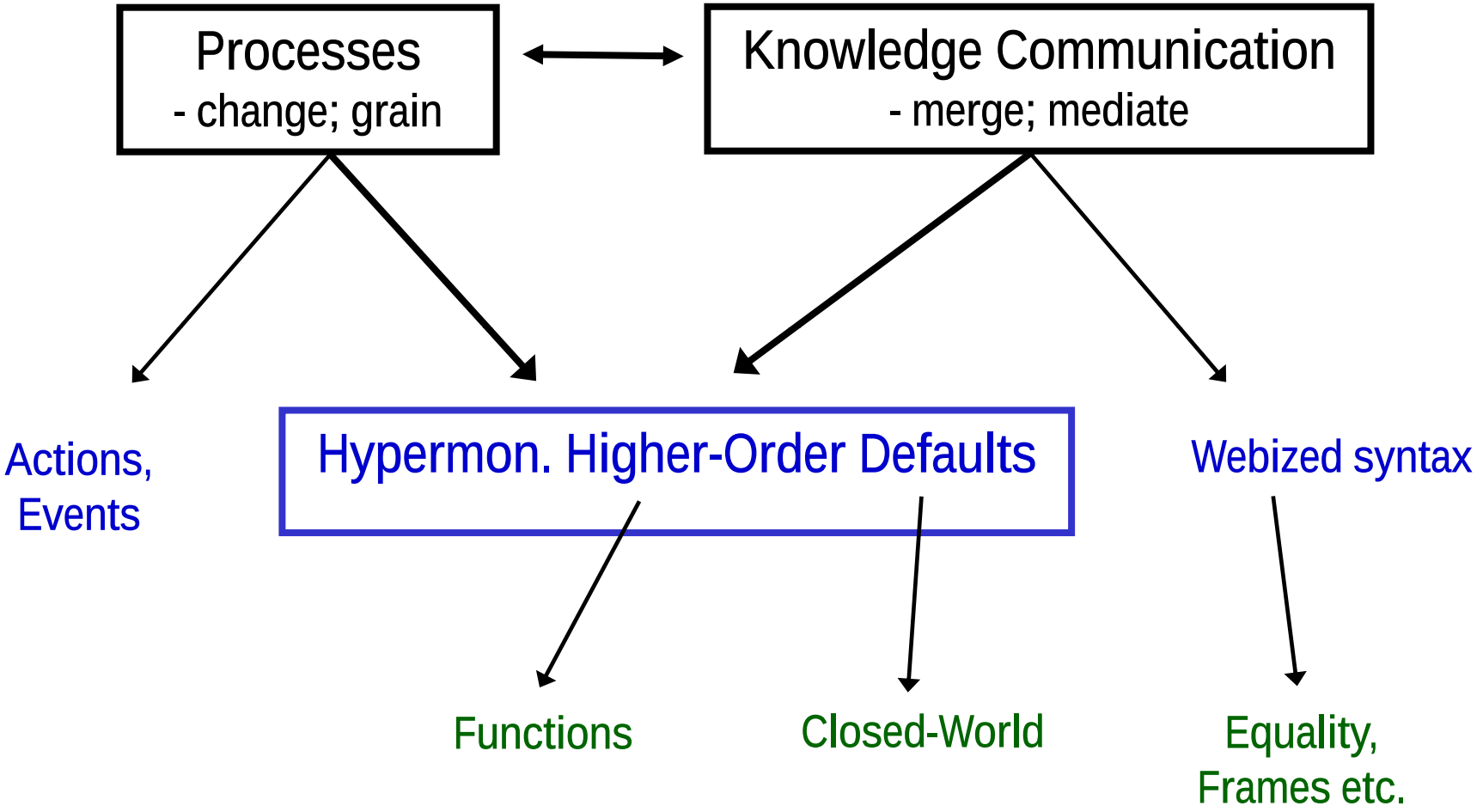


PROJECT HALO

VULCAN

34

Dependencies among Requirements IV



PROJECT HALO

VULCAN

35

Strategy on Expressiveness

- That's a Lot! Can We Do It? How?
  - Where to Start?
  - How to Factor?

- 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

Outline of Talk

- Overview
  - Vision, Origins, Goals, Effort, Approach, Roots, Status
  - V1 Prototype, Theory, Language; V2 plans
  - Examples and Use Cases

- Drill down on the KR Language and System
  - Requirements analysis
  - Hyper Logic Programs KR approach and expressive features
    - Higher-Order Defaults. 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

Hyper Logic Programs

- SILK uses a new KR: Hyper Logic Programs (HLP)
  - "Hyper" since it's Web (*hyper*text) centric, *hyper*monotonic, & *higher*-order.
- It integrates several major LP extensions never previously combined:
- Higher-order and Frames, cf. Hilog and F-Logic
- + Defaults, cf. Courteous LP (and Defeasible Logic)
  - Newly generalized and modified approach
  - Enables higher-order defaults
  - Implemented in SILK V1
- + Weakened Classical Logic, cf. *Hypermonotonic* mapping
  - New approach to semantic interchange of LP defaults with classical logic
  - In development for SILK V2
  - Background: *Hypermonotonicity* of an LP KB means that it is
    - nonmonotonic; and
    - sound but incomplete relative to a corresponding classical KB

PROJECT HALO

VULCAN

Hyper Logic Programs, continued

- 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 V2 Preview: 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

SILK V2 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 "Glass Mountain" – 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

Escape from Glass Mountain

- From the classic European fairy tale “The Princess on the Glass Hill”, in *The Blue Fairy Book*, by Andrew Lang, illustrated by Frank Godwin

PROJECT HALO

VULCAN

Outline of Talk

- Overview
  - Vision, Origins, Goals, Effort, Approach, Roots, Status
  - V1 Prototype, Theory, Language; V2 plans
  - Examples and Use Cases

- Drill down on the KR Language and System
  - Requirements analysis
  - Hyper Logic Programs KR approach and expressive features
    - Higher-Order Defaults. 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.

46

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.

47

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

49

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

50

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													
	We													
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.

52

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.

53

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

PROJECT HALO

VULCAN

Outline of Talk

- Overview
  - Vision, Origins, Goals, Effort, Approach, Roots, Status
  - V1 Prototype, Theory, Language; V2 plans
  - Examples and Use Cases

- Drill down on the KR Language and System
  - Requirements analysis
  - Hyper Logic Programs KR approach and expressive features
    - Higher-Order Defaults. 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

Reflections on Halo

- Halo is one of the most ambitious "classic AI" R&D programs in the US
  - We bring together graduate students, research labs, and universities into a unified, ambitious project
  - Halo is known worldwide
- 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

57

SILK – Recap

- A KR Language and KR System with reasoner, UI, interchange
- Goal: Expressiveness + Semantics + Scalability + Web
- Focus: Defaults and Processes
- Hyper LP KR combines new features
  - Defaults and Weakened Classical, cf. generalized Courteous LP
  - 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, and theory for expressive heart
  - V1 adds Higher-Order Defaults to Flora
  - Extensive requirements analysis, use cases, benchmarking; ReCyc translation
  - V2 in development

PROJECT HALO

VULCAN

SILK – Recap, continued

- 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 Mountain – yet retain grade-AAA model-theoretic semantics

PROJECT HALO

VULCAN

Future Directions for SILK

- Process – more complex
- Natural Language KA and UI
- Parallelism in reasoning
- Connectors to Semantic Web, legacy BRMS and DBMS
- Uncertainty
- Disjunction
- And Use Cases, of course

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

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

PROJECT HALO

VULCAN

Why the Sem. Tech. Industry Needs something like SILK (repeat)

- Need to raise abstraction level, e.g., for SME and NL KA/UI
- Need robustness & meta-reasoning for web KB integration
  - Cope with conflict, mediation, context, knowledge quality
  - Defaults $\Rightarrow$ robustness, modularity $\Rightarrow$ scalability
  - Higher-order $\Rightarrow$ 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

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.projects.semwebcentral.org
  - Mailing list: silk-announce@semwebcentral.org (very low volume)
- Provide comments on SILK language design
  - Initial public draft in ~ fall 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
  - V1 public release in ~ fall 2009; V2 in 2010
  - Also SMW+ upcoming release will have simple semantic LP rules of SILK-y flavor
    - In ~ fall 2009. Limited to Horn.

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 –
What the next generation Web will be spun from

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.