New Wave KRR:
Rulelog-based Deep Reasoning with NLP for Financial Regulatory Compliance and More

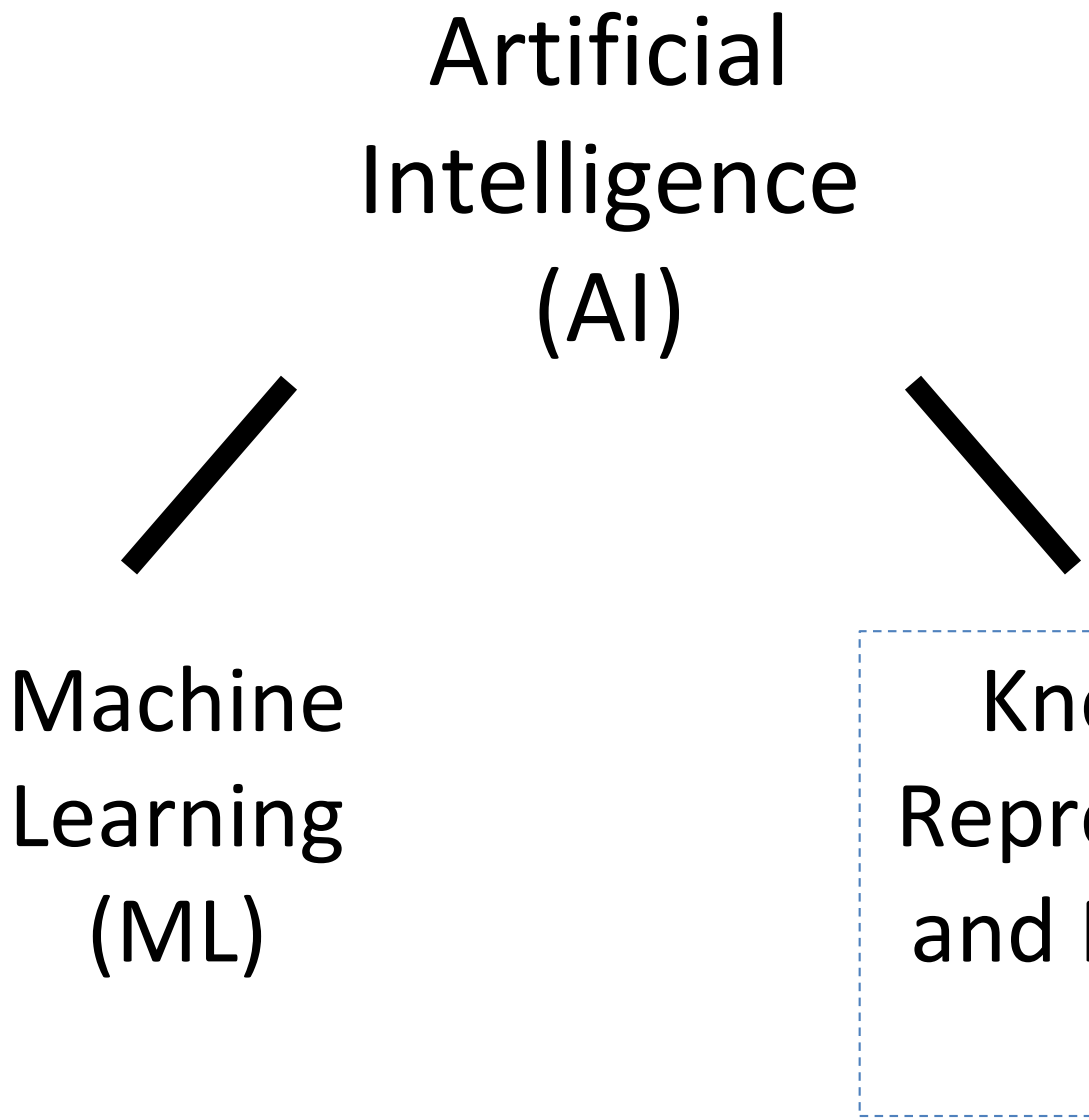
Benjamin Grosof*
Coherent Knowledge**

Presentation (30-min.)
for Industry Day
July 15, 2016
at International Joint Conference on Artificial Intelligence (IJCAI-16)
Held New York, NY, USA

** Web: http://coherentknowledge.com
* Email: *firstname dot lastname* at coherentknowledge.com

© Copyright 2016 by Coherent Knowledge.

The Core of AI



Logic-based KRR's Roles in AI

- Complements ML … in sense of induction from data
  … to enable ML in broader sense
- The power of cultural transmission
  - "Evolution's lesson" (Wolfgang Bibel)
- Accumulate knowledge coherently
- Communicate with humans: expertise, questions
- "Inject" ML results into predictable software

Coherent Knowledge: Company Overview

- Fundamentally new kind of logic/rule based AI software platform for advanced analytics: flexible deep reasoning + natural language processing
- Radical business benefits: accuracy/competence, cost, agility, transparency
- Company offers: software product Ergo + professional services for custom solution dev
  - Capabilities: engine + development environment, for executable knowledge bases (logic/rules) embedded in apps
- World-class founder team: created many industry-leading logic systems & standards
  - Extensive experience applying logic systems to financial, regulation/policy, and other domains
  - Former/current professors at Stony Brook University and MIT

Michael Kifer, PhD
Principal Engineer
Prof., Stonybrook Univ.
Winner, 3 ACM & ALP test-of-time research awards.

Benjamin Grosof, PhD
CTO & CEO
Prof., MIT Sloan. DARPA PI.
Advanced AI Prog. Mger. for Paul Allen.
Creator, IBM Common Rules.

Theresa Swift, PhD
Principal Engineer
Co-lead dev, mission-critical rules system, US Customs.
Co-Architect, XSB Prolog.

Paul Fodor, PhD
Senior Engineer
Prof., Stonybrook Univ.
IBM Watson team.

Janine Bloomfield, PhD
Director of Operations
Sr. Scientist, Climate Change, Environmental Defense Fund.
Data Science at Yale, US Forest Service.

Coherent Knowledge

Problem: Analytics for *Complex* Knowledge

Examples: policies, regulations, contracts; terminology mappings; causal pathways

Existing *Non-Semantic* Technologies tend to be:

- Shallow
- Siloed
- Costly, and Slow
- Patchily automated
- Opaque
- Inaccurate
- End users not empowered to modify

Based on:

- *Conventional programming languages*
- *Production/ECA rules*
- *Prolog*

Coherent Knowledge

Ergo is based on Textual Rulelog

- *Rulelog* is a kind of logical knowledge representation and reasoning (KRR)
  - A major research advance in KRR theory & algorithms, which culminated in 2012
- **Ergo is the most complete & highly optimized implementation available**
- Rulelog features very high/flexible expressiveness:
  - **Higher-order, general quantified formulas** (with logical chaining);
  - **Defeasibility (i.e., exceptions and argumentation)**;
  - Provenance, probabilistic, restraint bounded rationality, and more
- Yet Rulelog reasoning scales well: polynomial-time, as in databases
  - Millions of sentences concluded/asserted on a single processor
  - Up to trillions by orchestrating database etc. systems in distributed settings
- *Textual* Rulelog extends Rulelog with natural language processing (NLP)
  - Logic itself is utilized to map between English syntax and logic syntax
  - ErgoText templates aid knowledge entry and explanation generation

Series of Advances → Rulelog's Core Expressive Features

- Well-founded semantics; basic tabling algorithms
  - *Undefined* for paradox; smart cacheing; intuitionistic disjunction
- Higher-order syntax (Hilog); frame syntax
  - Associated optimizations of LP tabling etc. algorithms
- Statement id's for meta; argumentation meta-rules for defeasibility; provenance
- General formulas with all usual classical connectives and quantifiers (omniformity)
- Restraint bounded rationality
  - Use 3rd truth value *undefined* for "don't-care"
  - Radial, skipping; naf unsafety; external-query unsafety, unreturn

KRR Features Comparison: Rulelog Shines

Feature \ System	Rulelog Rules - e.g., Ergo	Datalog Rules - e.g., Jena, SWRL, Ontobroker, SPIN	Production Rules - e.g., IBM, Oracle, Red Hat	Prolog - e.g., SICStus, SWI, XSB	FOL & OWL-DL - e.g., Vampire, Pellet, Prover9	ASP Solvers - e.g., DLV, CLASP
Semantic & on standardization path	✓	✓	restricted case	restricted case	✓	✓
Basic expressiveness						
• Datalog LP	✓	✓	✓	✓	✓	✓
• Logical functions	✓	✗	✗	✓	✓	restricted
• Quantified formulas (genl.)	✓	✗	✗	✗	✓	✗
Full Meta expressiveness						
• Higher-order syntax, provenance	✓	✗	✗	✗ (except XSB a little)	✗	✗
• Defeasibility & well founded negation	✓	✗	✗	✗	✗	some have restricted
• Restraint bounded rationality	✓	✗	✗	✗	✗	✗
• Probabilistic	✓	✗	✗	✗	✗	✗
Efficiency						
• Goal-directed	✓	✗ (except Jena)	✗	✓	✓	✓
• Full LP tabling with dependency-aware updating	✓	✗	✗	✗ (except XSB)	✗	✗
• Polynomial time complexity	✓	✓	✓	✗	✗	✗

Notes on KRR Features Comparison

- "System" means system type / approach of logical knowledge representation and reasoning (KRR).
- "Semantic" means in the sense of KRR, i.e., fully declarative and having a model theory in the logical sense.
- "FOL" means First Order Logic. "ASP" means Answer Set Programs.
  - ASP is recently emerging. The tasks for which it's suitable are more similar to FOL than to the other systems here.
- "Standardization" here means industry standardization. "On path to" means in process of being, or already, standardized.
- "Restricted case" means for a syntactic/expressive subset.
- Event-condition-action rules in this context are similar to, and lumped in with, production rules.
- "LP" means declarative logic programs.
- Datalog means LP without logical functions. Usually this is restricted to Horn. But here we permit negation(-as-failure).
- OWL-RL is pretty much a restricted case of Datalog LP.
- "Higher-order syntax" means Hilog, which enables probabilistic – and also 1) fuzzy and 2) frame syntax cf. F-Logic.
- "Provenance" means provenance info about assertions, via properties of rule id's that are within the logical language / KRR.
- "Full" applies to all four of the meta expressiveness features.
- Defeasibility includes flexible argumentation theories.
- "General formulas" means classical-logic-like formulas, including with head existentials and with head disjunction.
- "LP tabling" includes sophisticated: cacheing of intermediate reasoning results, inference control, and indexing.
- "Dependency-aware updating" means that when assertions are added or deleted, saved inferences are only recomputed if they depend on the changes to the assertions.
- Polynomial time "complexity" means worst-case computational complexity, with constant-bounded number of variables per rule. Polynomial-time is similar to database querying, and is a.k.a. "tractable".

- Datalog X defeasibility: Ontobroker has full well founded negation.
- Prolog X defeasibility: XSB has full well founded negation.
- ASP X defeasibility: Some ASP systems have restricted defeasibility & well founded negation. ASP systems essentially have wf negation inside (i.e., as part of) their semantics/reasoning, and some ASP systems even expose it to the user.
- Datalog X goal-directed: Jena has a backward engine as well as a forward engine.
- ASP X general formulas: ASP has head disjunction.
- FOL X full LP tabling with dependency-aware updating: Some FOL theorem-provers cache intermediate results in a way that is analogous to LP tabling, and some do dependency tracking but we're not sure how analogous or sophisticated.
- Prolog X higher-order syntax: XSB has some support for this (i.e., for Hilog), although it is not integrated well.

Ergo Architecture

Optional Custom Solutions

Ergo Suite

Ergo Studio

Rule Editor and Query UI

(Integrated Development Environment)

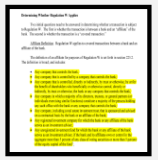
Ergo Reasoner

Knowledge Base

queries, assertions

answers, explanations

Complex Information
- English Doc.'s etc.
- Policies, Regulations
- Financial, Legal, Science



Users

API's Java WS

API's Java WS

External Services & Frameworks

- Relational DB
- RDF/Graph DB
- Other Sem. Tech
- Machine Learning
- Apps, Docker, …

External Info (multi-source)

- Data
- Views, Rules
- Schemas & Ontologies
- Results of ML

events, decisions

App Actions

WS = Web Services. Sem. = Semantic. ML = Machine Learning

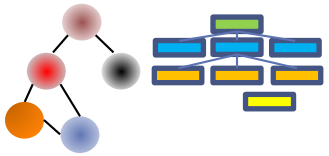
© Copyright Coherent Knowledge Systems, LLC, 2015. All Rights Reserved.

Ergo Suite: Reasoner, Studio, Connectors

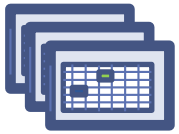
- Ergo Reasoner has sophisticated algorithms & data structures
  - Smart cacheing with dependency-aware updating. Leverages LP & DBMS techniques.
  - Transformation, compilation, reordering, indexing, modularization, dependency/loop analysis, performance monitoring/analysis, pausing, virtual machine, programming kernel, external import/querying
  - Java API. Other interfaces: command line, web, C.
  - Scales well: Millions of sentences on 1 processor; Trillions on distributed nodes
- Ergo Studio is a graphical Integrated Development Environment
  - Interactive editing, querying, explanation, visualization of knowledge
  - Fast edit-test loop with award-winning advanced knowledge debugging/monitoring
- Ergo Connectors federate knowledge & reasoning
  - Import/query dynamically via: SPARQL, OWL, RDF; SQL; CSV; JSON; and more
  - Federation distributes reasoning (i.e., its processing) across multiple nodes
- Open, standards-based approach; a portion is open source
  - Rulelog is draft industry standard from RuleML (submission to W3C & Oasis)

ERGO works with existing data sources, at scale

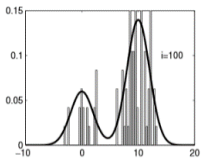
NoSQL DB, OWL, & Knowledge Graph



Relational DB



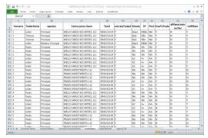
Machine Learning & Stats



$Prob_N(x) = \Sigma_{i=1..N} \delta_{si}(x) / N$

Domain Apps & Legacy

Spreadsheet



"Business Rules"

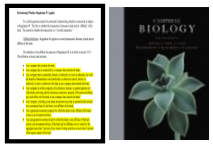
$tier(X,1) \wedge supply(Y,X) \Rightarrow tier(Y,2)$

Probabilistic engines

$p(H|C) = \Pi\, p(x_i|Y_j) / \Sigma_C \Pi\, p(x_i|Y_j)$

External Info & Services

Text & Natural Language processing



Queries
Assertions
Edits

ERGO

Answers, Views
Decisions, Alerts
Explanations

Coherent Knowledge

Case Study: Automated Decision Support for Financial Regulatory/Policy Compliance

Problem: Current methods are expensive and unwieldy, often inaccurate

Solution Approach – using Textual Rulelog software technology:

- Encode regulations and related info as semantic rules and ontologies
- Fully, robustly automate run-time decisions and related querying
- Provide understandable full explanations in English
  - *Proof*: Electronic audit trail, with provenance
- Handles increasing complexity of real-world challenges
  - Data integration, system integration
  - Conflicting policies, special cases, exceptions
  - What-if scenarios to analyze impact of new regulations and policies

Business Benefits – compared to currently deployed methods:

- More Accurate
- More Cost Effective – less labor; subject matter experts in closer loop
- More Agile – faster to update
- More Overall Effectiveness: less exposure to risk of non-compliance

Demo of Ergo Suite for Compliance Automation: US Federal Reserve Regulation W

- EDM Council Financial Industry Consortium Proof of Concept – ***successful and touted pilot***
  - Enterprise Data Management Council (Trade Assoc.)
  - Coherent Knowledge (USA, Technology)
  - SRI International (USA, Technology)
  - Wells Fargo (Financial Services)
  - Governance, Risk and Compliance Technology Centre (Ireland, Technology)
- Reg W regulates and limits $ amount of transactions that can occur between banks and their affiliates. Designed to limit risks to each bank and to financial system.
- Must answer 3 key aspects:

1. *Is the transaction's counterparty an affiliate of the bank?*
2. *Is the transaction contemplated a covered transaction?*
3. *Is the amount of the transaction permitted ?*

The Starting Point - Text of Regulation W

Determining Whether Regulation W Applies

Two initial questions need to be answered in determining whether a transaction is subject to Regulation W. The first is whether the transaction is between a bank and an "affiliate" of the bank. The second is whether the transaction is a "covered transaction."

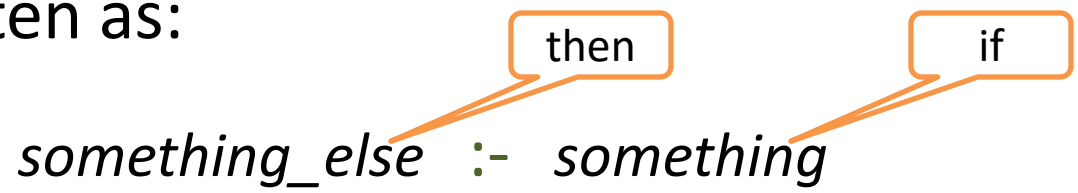
Affiliate Definition. Regulation W applies to covered transactions between a bank and an affiliate of the bank.

The definition of an affiliate for purposes of Regulation W is set forth in section 223.2. The definition is broad, and includes:

- Any company that controls the bank;
- Any company that is controlled by a company that controls the bank;
- Any company that is controlled, directly or indirectly, by trust or otherwise, by or for the benefit of shareholders who beneficially or otherwise control, directly or indirectly, by trust or otherwise, the bank or any company that controls the bank;
- Any company in which a majority of its directors, trustees, or general partners (or individuals exercising similar functions) constitute a majority of the persons holding any such office with the bank or any company that controls the bank;
- Any company, including a real estate investment trust, that is sponsored and advised on a contractual basis by the bank or an affiliate of the bank;
- Any registered investment company for which the bank or any affiliate of the bank serves as an investment adviser;
- Any unregistered investment fund for which the bank or any affiliate of the bank serves as an investment adviser, if the bank and its affiliates own or control in the aggregate more than 5 percent of any class of voting securities or more than 5 percent of the equity capital of the fund[1];

Ergo Makes Sentences Executable

- If *something* is true then *something else* must be true. Written as:



- Example of executable Ergo sentence:

\(The individual affiliate threshold for transaction under Regulation W
by ?Bank with ?Counterparty is ?Amount\) :-

\(?Counterparty is deemed an affiliate of ?Bank under Regulation W\) \and
\(?Bank has capital stock and surplus ?Capital\) \and
\(the threshold percentage for an individual affiliate is ?Percentage\) \and
?Amount = ?Capital * ?Percentage/100.

Demo goes here

- Note: Several screenshots are in the backup slides.

Demo Summary

- Encode regulations and related info as semantic sentences in human-machine logic (rules/facts with logic tied closely to English)
- Fully, robustly automate run-time decisions and related querying
- Provide understandable full explanations in English
  - *Proof*: Electronic audit trail, with provenance
- Not shown due to limited time:
  - Handle exceptions, special cases, conflicting policies
- Handle increasing complexity of real-world challenges
  - Data integration, system integration

ErgoText

- ErgoText:

\(The proposed transaction ?Id by ?Bank with ?Affiliate of $?Amount is a RegW covered transaction\)

- ErgoText Template:

```
template(headbody,
            \(The proposed transaction ?Id by ?Bank with ?Affiliate of $?Amount
              is a RegW covered transaction\),

           covered(proposed(transaction))(by(?Bank))(with(?Affiliate))
             (of(amount(?Amount)))(having(id(?Id)))
           ).
```

- The templates are self-documenting

Textual Rulelog (III)

- Almost any NL sentence can be represented as a logical sentence
  - Leverages the logical quantifiers feature of Rulelog
  - Ex.: "each large company has some talented CEO"
    - forall(?x)^((?x \isa \(large company\)) ==>
      exists(?y)^((\?x has ?y\) \and
      (?y \isa \(talented CEO\)))).

Benefits of this Approach to Analytics

- Deep in reasoning & knowledge
- Accurate
- Transparent, with explanations
- More Cost-Effective & Agile
- More Automated
- Easy to modify,
  end users empowered
- Greater Integration
- Greater Reusability
- What-if analyses

Analyze

Explain

Make Decisions

Monitor & Alert

Answer Questions

Application Areas for Rulelog (I)

- *Commercially, to date:*
- Financial regulatory/policy compliance
- Defense intelligence analysis
- Info integration: defense, financial, supply chain
- E-commerce pricing/promotion policies

Application Areas for Rulelog (II)

- *Explored in research, to date*
  … & promising commercially as further sub-areas
- Confidentiality policies: security, social media, HIPAA
- Financial/business reporting: XBRL
- Contracts: e-commerce, financial instruments, license agreements, large construction
- Health: treatment guidance, insurance

Application Areas for Rulelog (III)

- *Explored in research, to date*
  … & promising commercially further areas
- Education/e-learning: personalized tutoring
- NL understanding and conversational interfaces
- Workflow / business process management: helpdesk, personal communications

Intelligent Assistants and NL Understanding

- *The Promise:* Able to converse with and assist humans in many facets of our lives
  - Provide advice
  - Perform tasks
  - Inform us proactively
  - Explain why

- *Required to fulfill this promise:* Flexible deep reasoning
  - Using logical/probabilistic knowledge representation and reasoning (KRR)
  - Combined with natural language processing (NLP) and machine learning (ML)
  - Treat the **deep** semantics of NL
  - KRR was central to first wave of AI success. KRR + ML = core of AI.
  - KRR is cognitive computing's weakest leg today (ML > NLP > KRR)

Textual Rulelog Interpretation (TRI)

- Use Textual Rulelog to do text interpretation,
  i.e., interpret broad NL syntax into logic syntax
- Textual entailment (TE) via logical entailment (LE) in Rulelog:
  1. Syntax Interpretation (SI): 1-by-1, map NL sentences into logic sentences
     - This can be viewed as transformation but is actually implemented in Rulelog itself via an system-internal application of logical inferencing.
  2. Reason logically to infer more logic sentences (as believed conclusions)
  3. Then, 1-by-1, map logic sentences into NL sentences
     - This is much easier than (1.).
- Summary: TE = SI + LE

- Focus for technology path:
  - Syntax Interpretation, esp. (logical) quantification
  - Utilize NL parsing and closely related NLP tools, plus UI for humans amending
  - Represent in Textual Rulelog

Coherent Knowledge

Combining ML and KRR

- Core AI = KRR + ML. KRR is required for Cognitive Computing too.
- The prediction step of ML requires reasoning
- The target of ML is a representation
- Getting value from ML requires reasoning for analysis and decisions
- KRR is required to combine results of ML, accumulate knowledge coherently, and explain knowledge
  - Weaknesses of ML today
- Reasoning to supply derived facts for ML to chew on
- Reasoning to focus ML's tasks and conjecture schemas
  - e.g., sets of relevant features, important questions, to drive ML

Ergo Strengths (I)

- Represent more complex knowledge – encode & utilize it
  - Almost any sentence articulable in English natural language
  - Policies, regulations, contracts, causal pathways, science
  - Terminology mappings, and context, for data and system integration
  - The actual questions one wants to ask
  - *Capture & inject subject matter experts' (SMEs') insights, directly*

- Reason deeply – assemble & compose multiple analysis results
  - Many steps. Prioritize and weigh counter-arguments.
  - Orchestrate multiple knowledge sources & components
  - Supports high accuracy

Ergo Strengths (II)

- Explain each answer – fully yet understandably
  - Every logical step is available, and described in English natural language
  - Interactively browsable – user chooses drill downs

- Overall: *modeling* instead of *programming*
  - Faster, cheaper, more reusable

For more info

- IJCAI-16 half-day tutorial on Rulelog KRR

- http://coherentknowledge.com ,
  incl. -> Publications and Presentations page

Thank you.

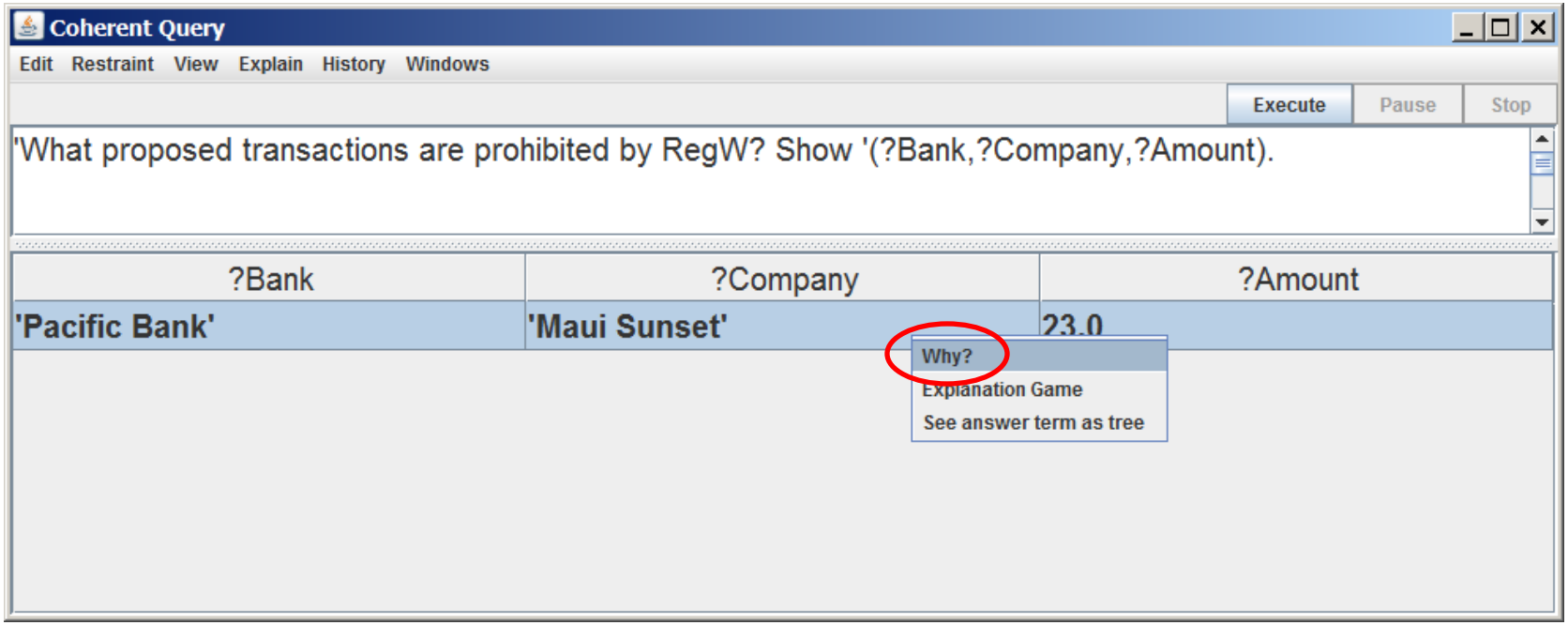
Deep Reasoning for Advanced Analytics

http://coherentknowledge.com

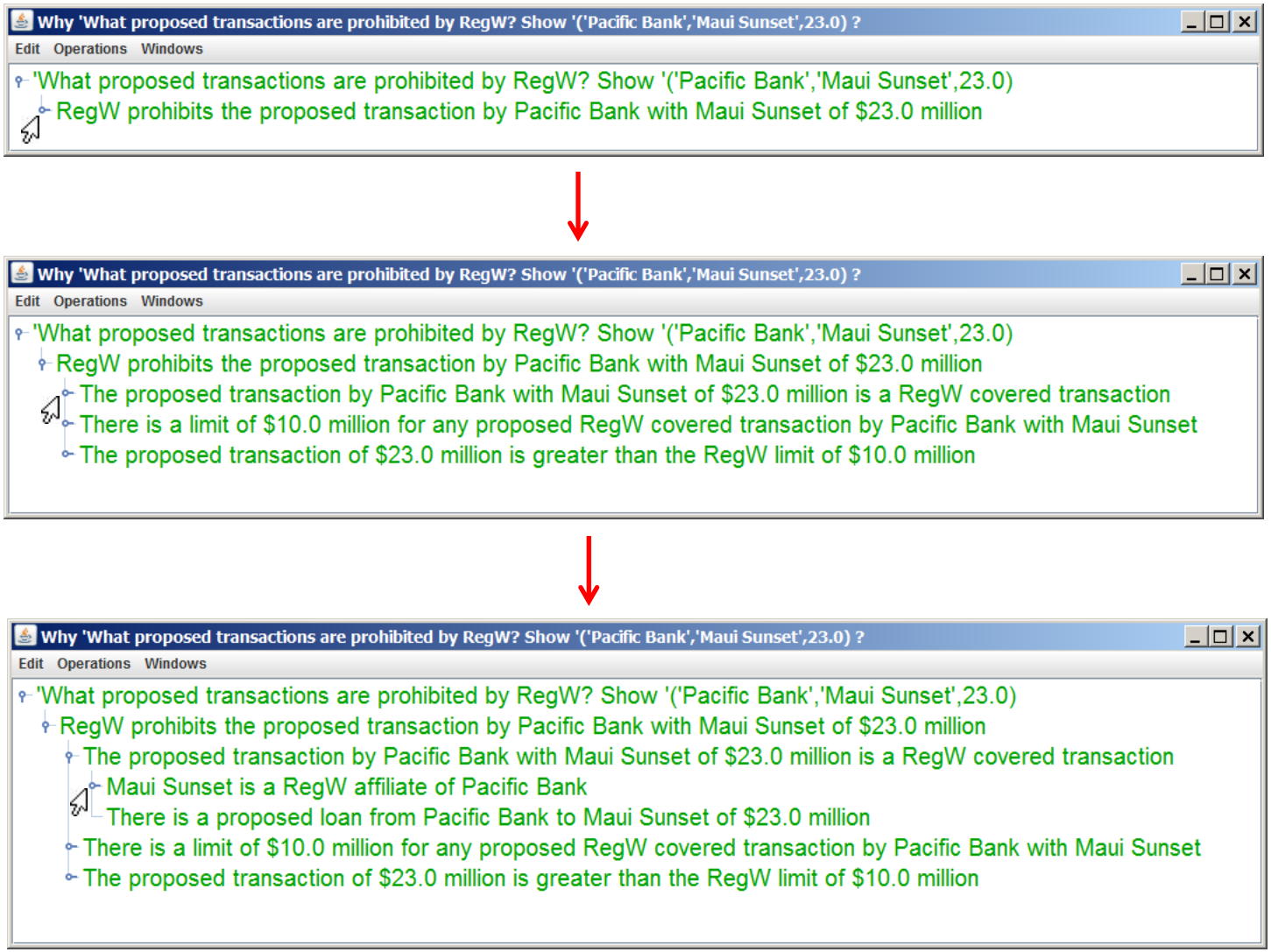
Disclaimer: All brands, logos and products are trademarks of their respective companies.

OPTIONAL/BACKUP SLIDES FOLLOW

Query is asked in English



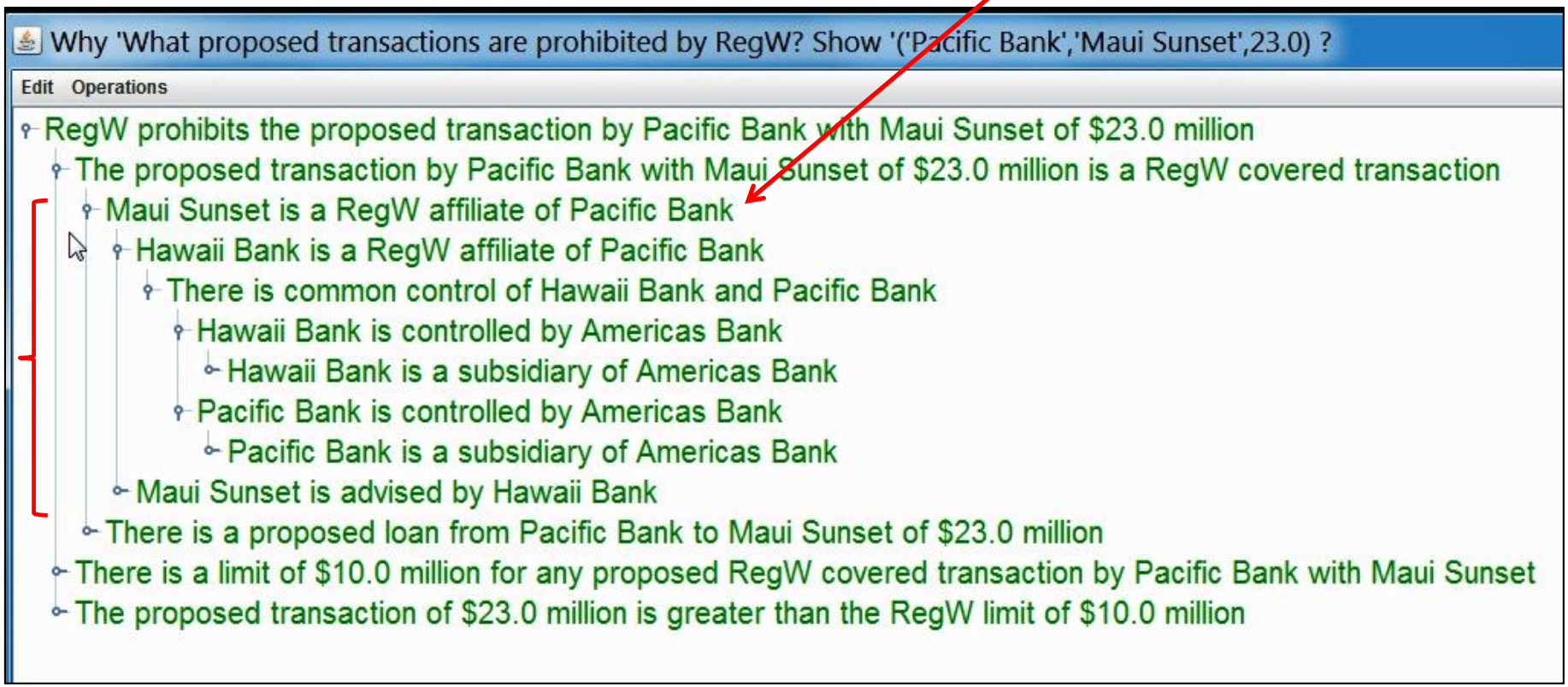
User Clicks the handles to expand the Explanations



Why is the proposed transaction prohibited by Regulation W?

1. *Is the transaction's counterparty an "affiliate" of the bank?*

YES.



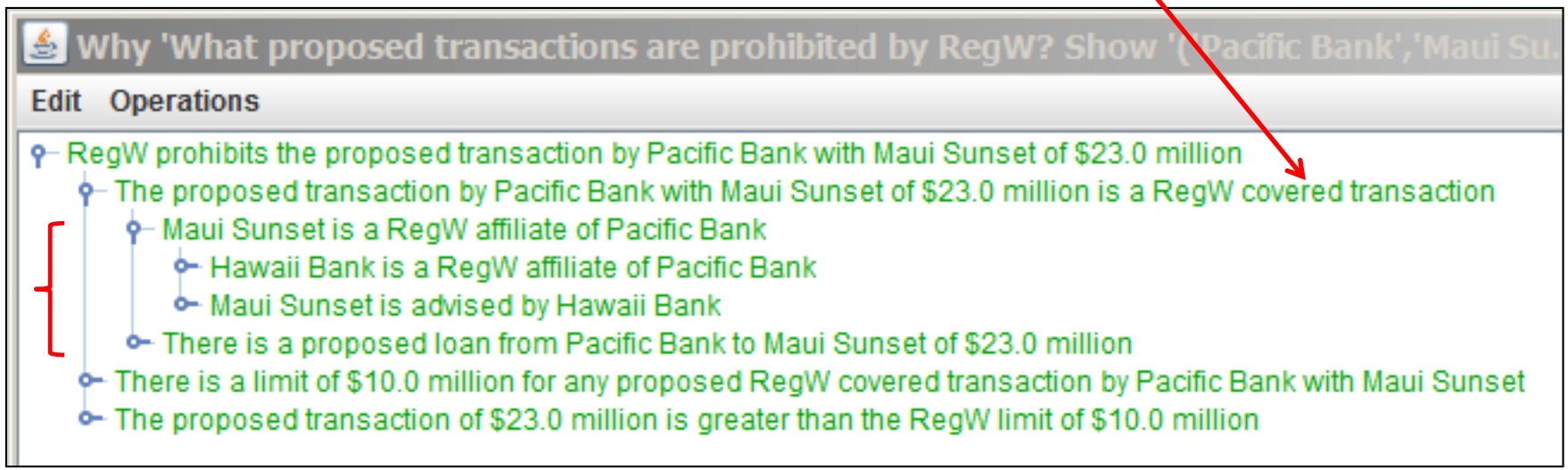
And here's why …

Why is the proposed transaction prohibited by Regulation W?

2. *Is the transaction contemplated a "covered transaction"?*

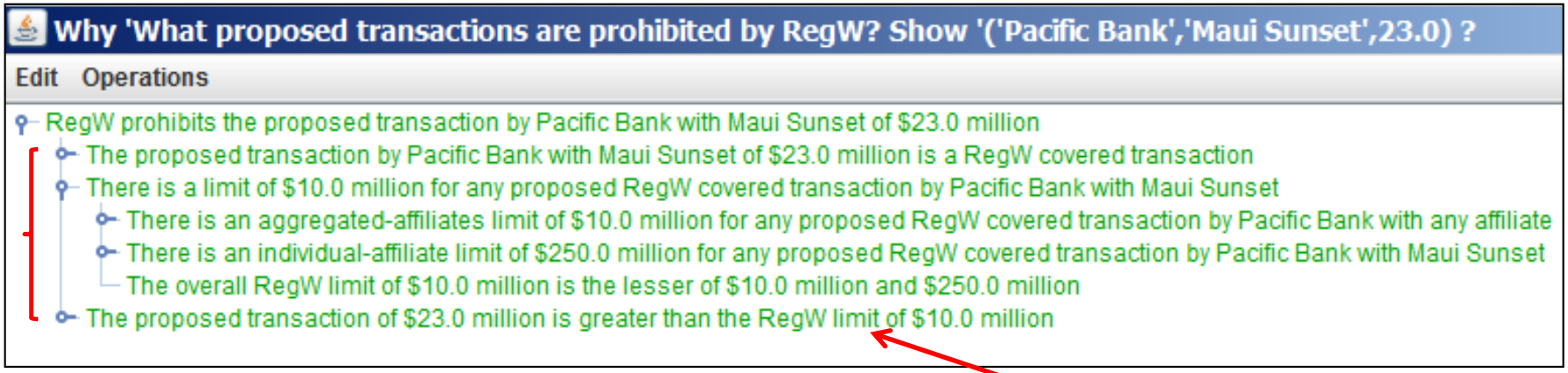
YES.

And here's why …



Why is the proposed transaction prohibited by Regulation W?

3. *Is the amount of the transaction <u>permitted</u>?*

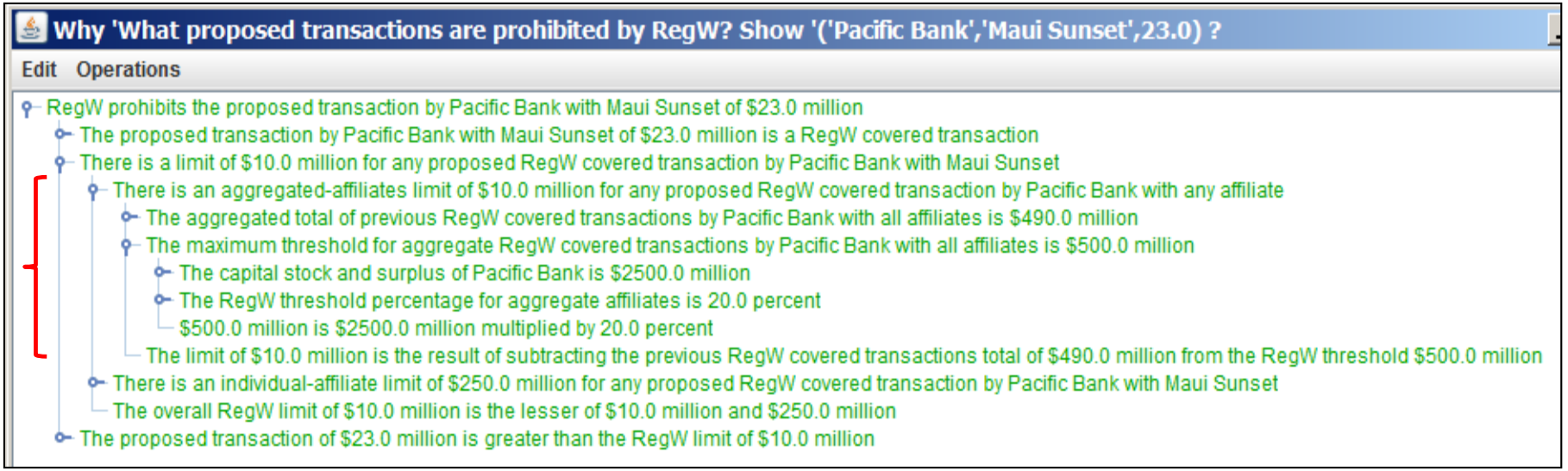


And here's why …

NO.
It went over the limit.

Why is the proposed transaction prohibited by Regulation W?

3. (continued) Why is the aggregate-affiliates limit $10 million?



Bio

- AI researcher, turned entrepreneur
- CTO, CEO, Co-Founder, of Coherent Knowledge
  - AI software platform component startup
- Previously:
  - Directed advanced AI research program for Paul Allen
    - Developed Rulelog KRR theory, algorithms, UI approach
  - MIT Sloan professor and DARPA PI
    - Co-Founder of RuleML, key contributor to W3C OWL-RL and RIF standards
  - IBM Research, creator IBM Common Rules
    - 1st successful semantic rules product in industry
  - Stanford AI PhD, combining ML with logical and probabilistic reasoning
- http://www.linkedin.com/in/benjamingrosof
- http://benjamingrosof.com

Coherent Knowledge

END of OPTIONAL/BACKUP SLIDES