

## A Hybrid Diagnosis Approach Combining Black-Box and White-Box Reasoning

Mingmin Chen<sup>1</sup>

Shizhuo Yu<sup>1</sup>

Nico Franz<sup>2</sup>

Shawn Bowers<sup>3</sup>

Bertram Ludäscher<sup>1,4</sup>

<sup>1</sup> Department of Computer Science , University of California, Davis

<sup>2</sup> School of Life Sciences, Arizona State University

<sup>3</sup> Department of Computer Science, Gonzaga University

<sup>4</sup> GSLIS & NCSA, University of Illinois at Urbana-Champaign

# Outline

- **The Taxonomy Alignment Problem**

$T_1 + T_2 + A \Rightarrow T_3$       (*ambiguous .. unique .. inconsistent*)

- **Model-based Diagnosis [Reiter'87]**

- Black-box

- **Hybrid Approach**

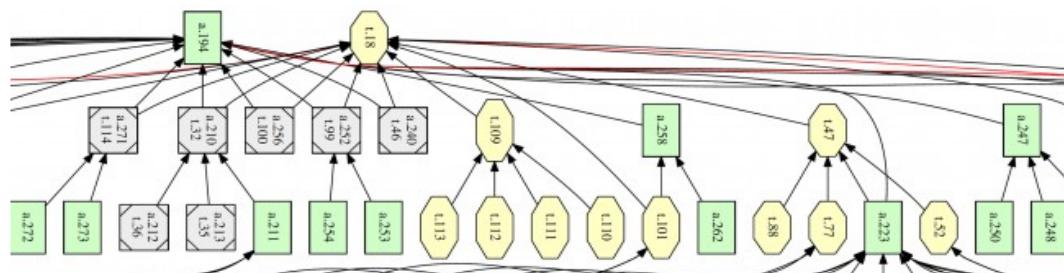
- Black-box & White-box combined

- **Benchmark Results**

# Meet Prof. Nico Franz: Curator of Insects @ ASU

**taxonbytes**  
from specimens to language, logic, and learning

Home Members Publications ASUHIC Informatics Outreach Links Contact



Initial alignment of two weevil classifications with the Euler toolkit

Linked here is an interim result of my attempt to align two influential weevil classifications by Thompson (1992) and Alonso-Zarazaga & Lyal

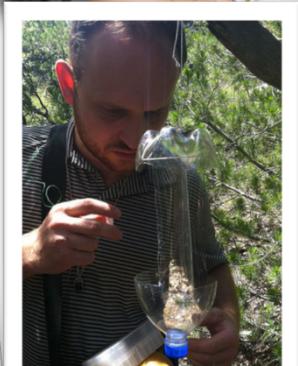
N. M. Franz / Cladistics 30 (2014) 294–321

296



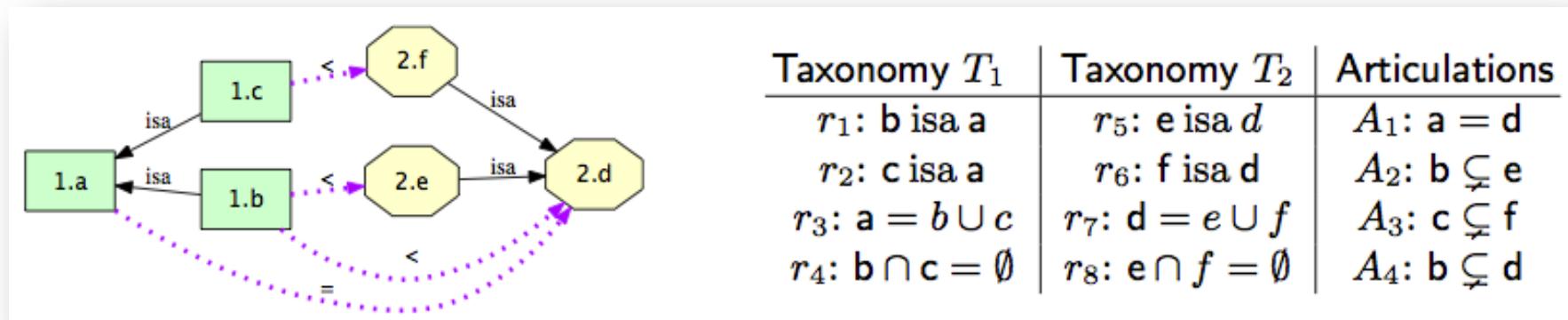
Hybrid Diagnosis in Euler @ RuleML2014, Prague

# What Nico et al. do for a living ...



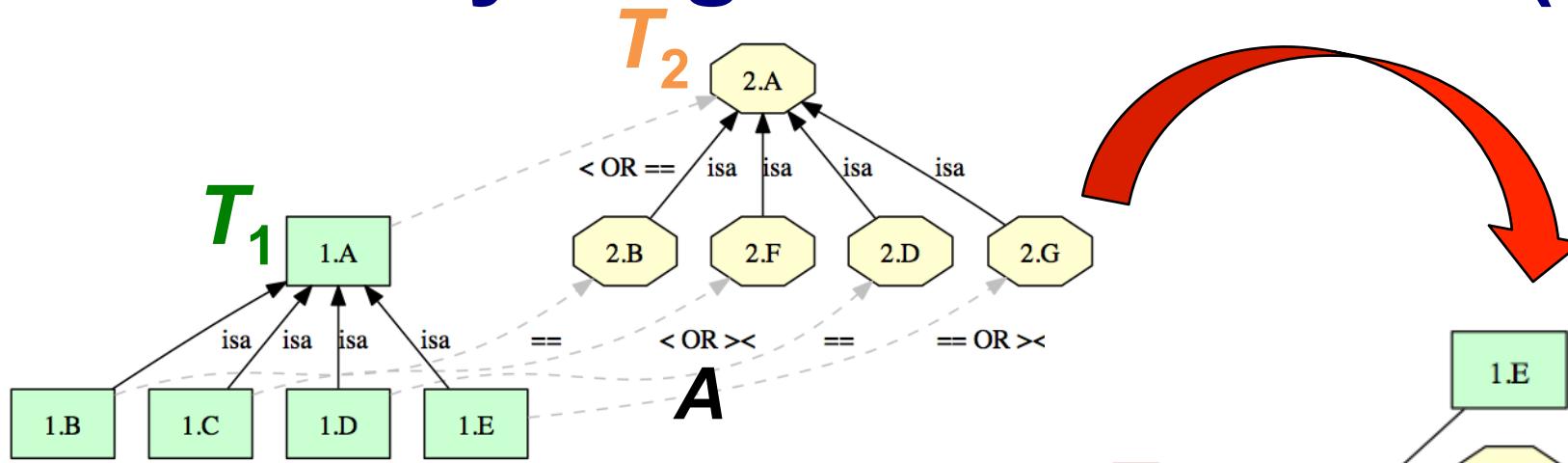
# What Nico does for a living (cont'd): The Indoors Part

- First: go fun places, find new bugs, study them ...
  - “Bugs-R-Us” (see **taxonbytes.org**)
- Now: Compare, **align** and **revise** taxonomies, based on careful observation, “character” data, expertise ...
- Formally:
  - Input:  $T_1$  +  $T_2$  (*taxonomies*) + A (*expert articulations*)

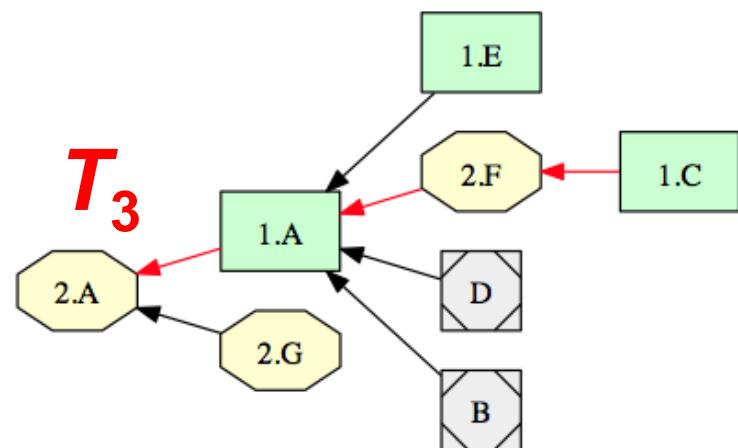


- Output: revised, “merged” taxonomy (-ies)  $T_3$

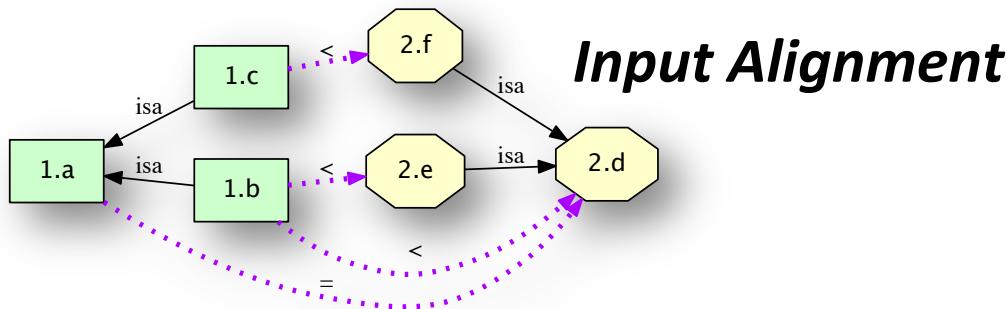
# Taxonomy Alignment Problem (TAP)



- **Given:**
  - Taxonomies  $T_1$ ,  $T_2$ 
    - incl. constraints (coverage, disjointness)
  - Set of articulations (*alignment*)  $A$
- **Find:**
  - Combined (“merged”) taxonomy  $T_3$  ( $= T_1 + T_2 + A$ )
    - Is it a taxonomy? Or a DAG?
  - Optional:
    - Final alignment (should be minimal)

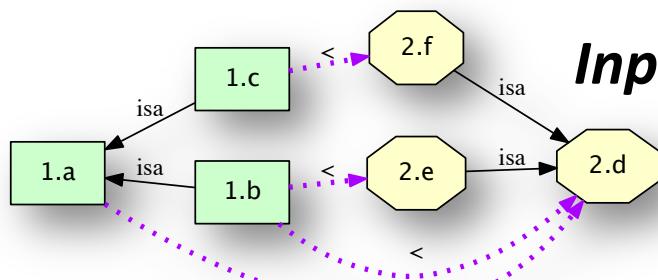


# TAP: Possible Outcomes



# TAP: Possible Outcomes

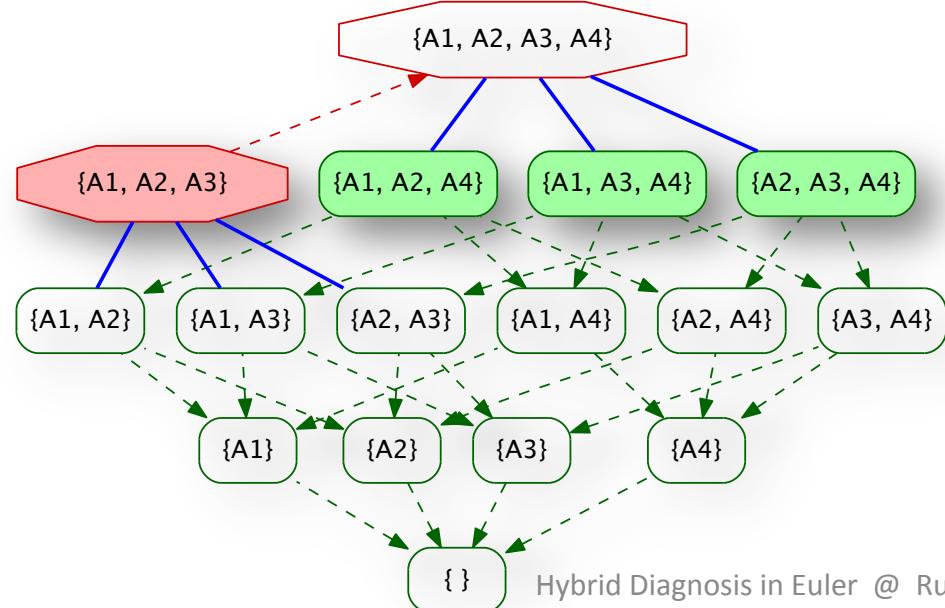
## *Input Alignment*



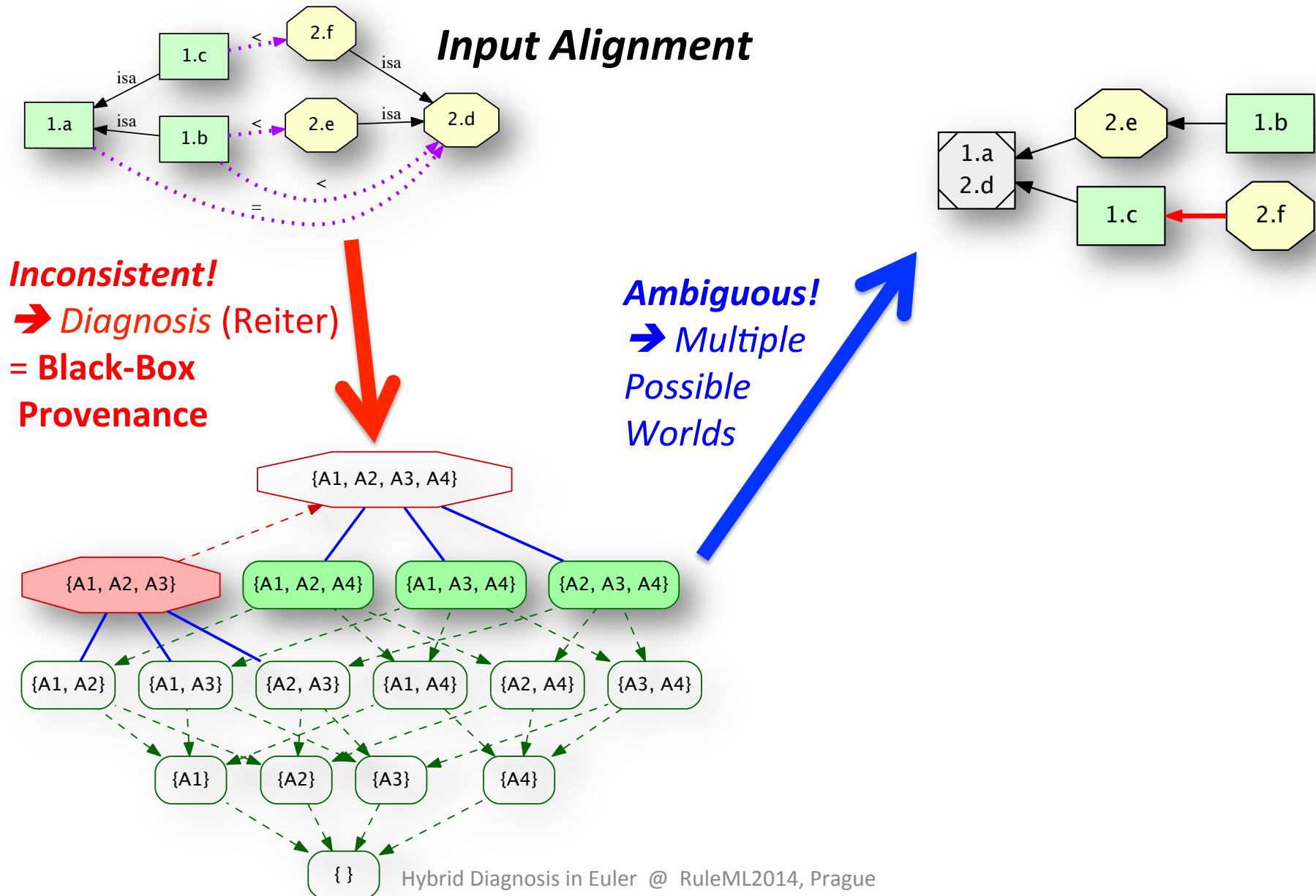
**Inconsistent!**

→ Diagnosis (Reiter)

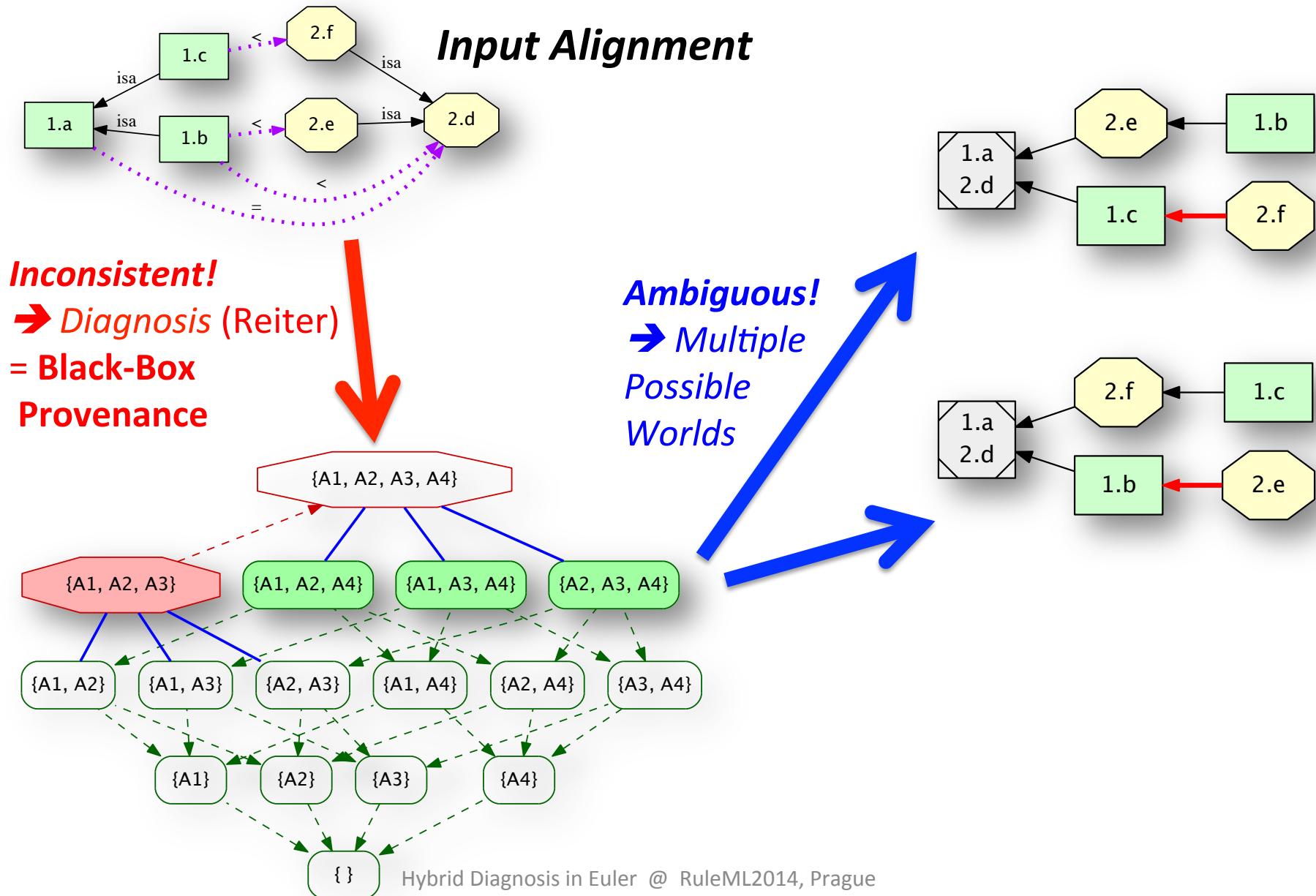
= Black-Box  
Provenance



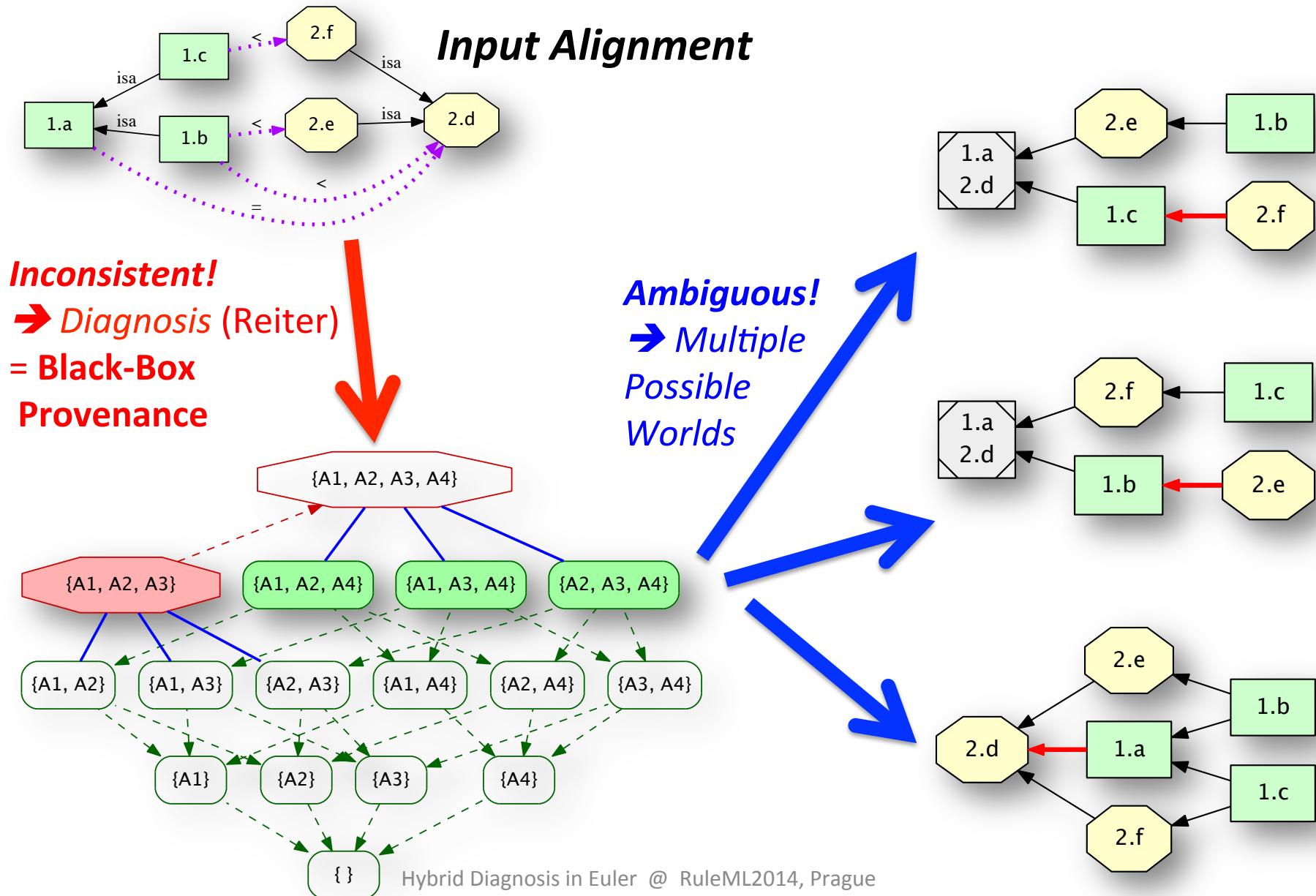
# TAP: Possible Outcomes



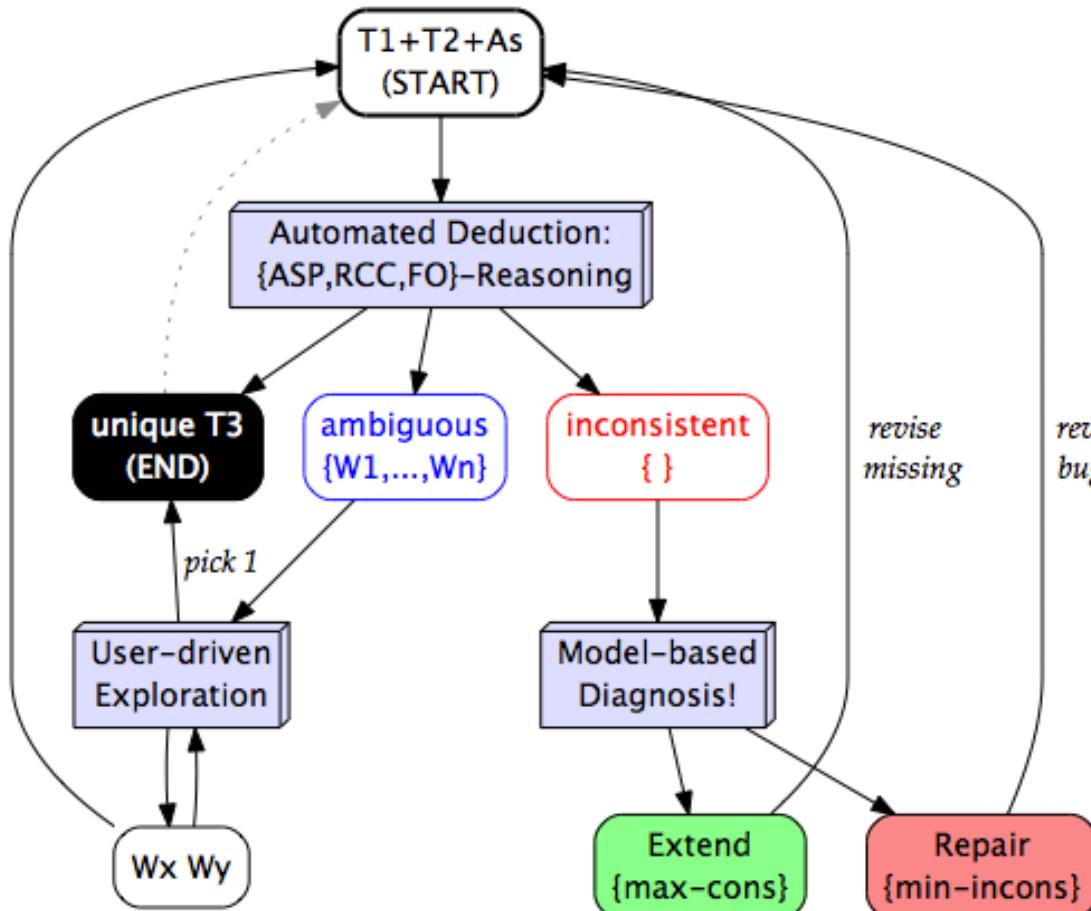
# TAP: Possible Outcomes



# TAP: Possible Outcomes



# Euler/X Toolkit and Workflow

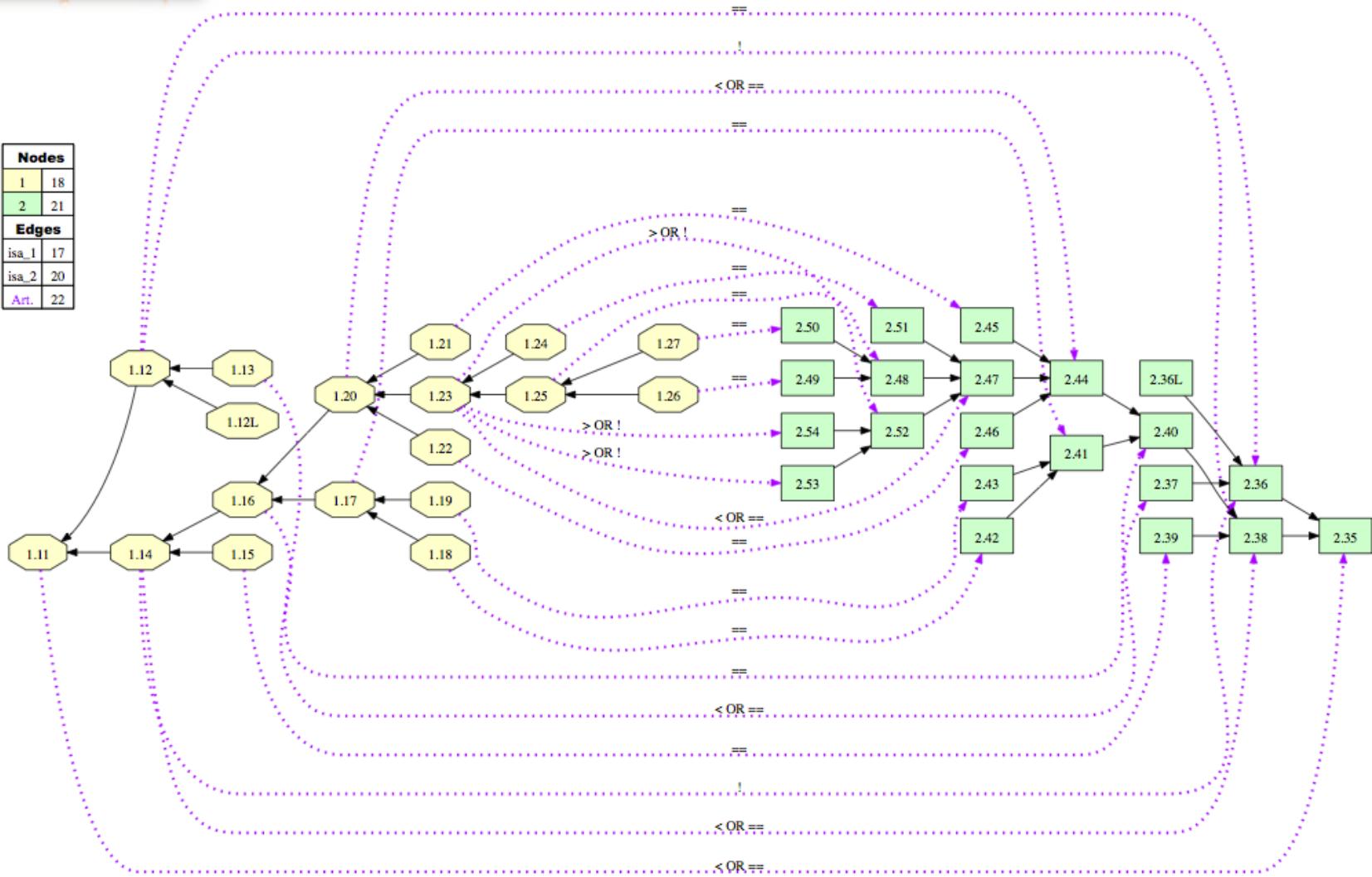


- **FO reasoning** about taxonomies (MFOL)
- Earlier: **CleanTax**
  - Prover9/Mace4
- Now: **Euler**
  - ASP Reasoners (DLV, Clingo)
  - Specialized reasoners (PyRCC)
  - ...
  - **X** = ASP, RCC, ...



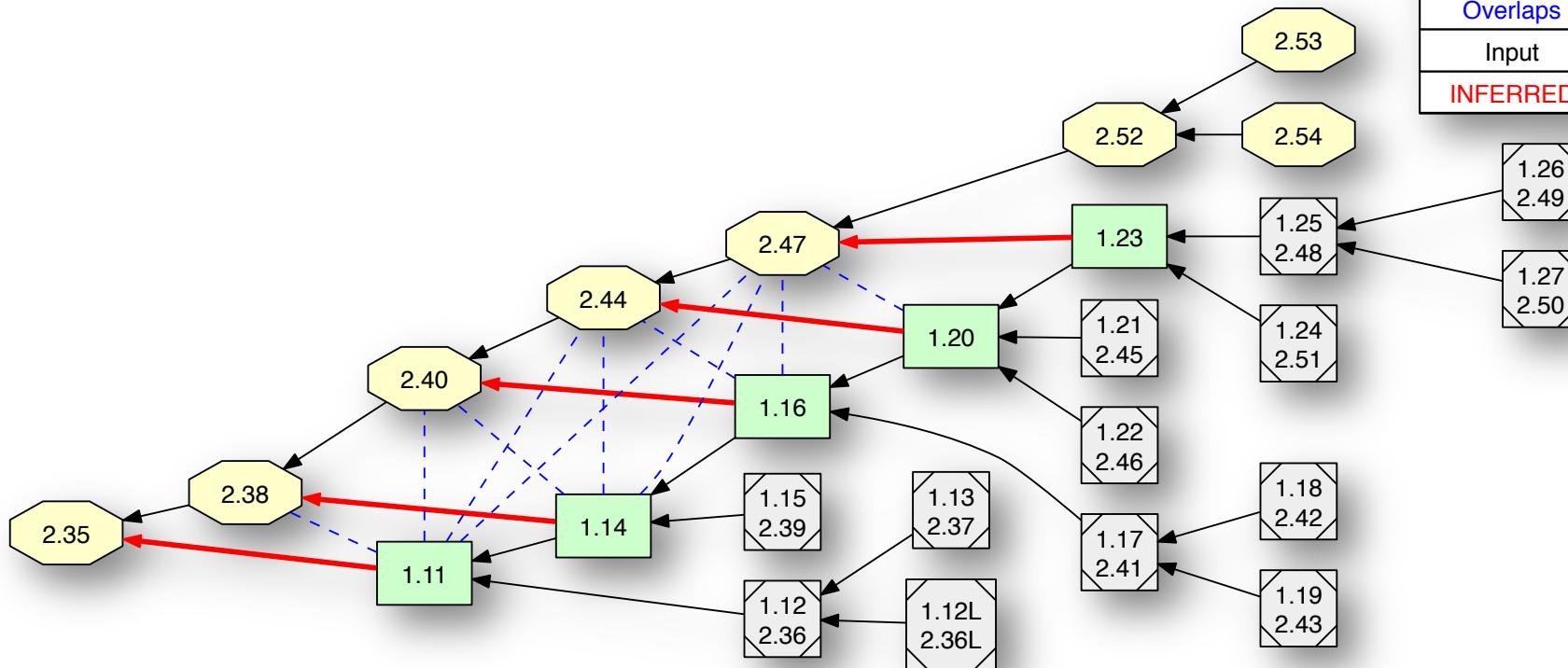
# Real-world examples: Turn this ...

Nodes	
1	18
2	21
Edges	
isa_1	17
isa_2	20
Art.	22



# ... into this!

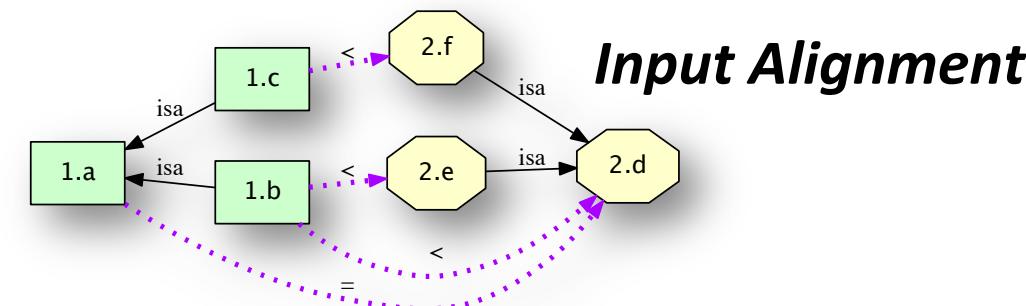
## (*Perellesscus* Alignment Result)



Nodes	
Taxonomy 1	5
Taxonomy 2	8
MERGED Taxa	13
Edges	
Overlaps	10
Input	24
INFERRED	5

- $T_3 := T_1 \text{ and } T_2$  are “merged”
  - Blue dashed: overlaps → resolve via “zoom-in view”

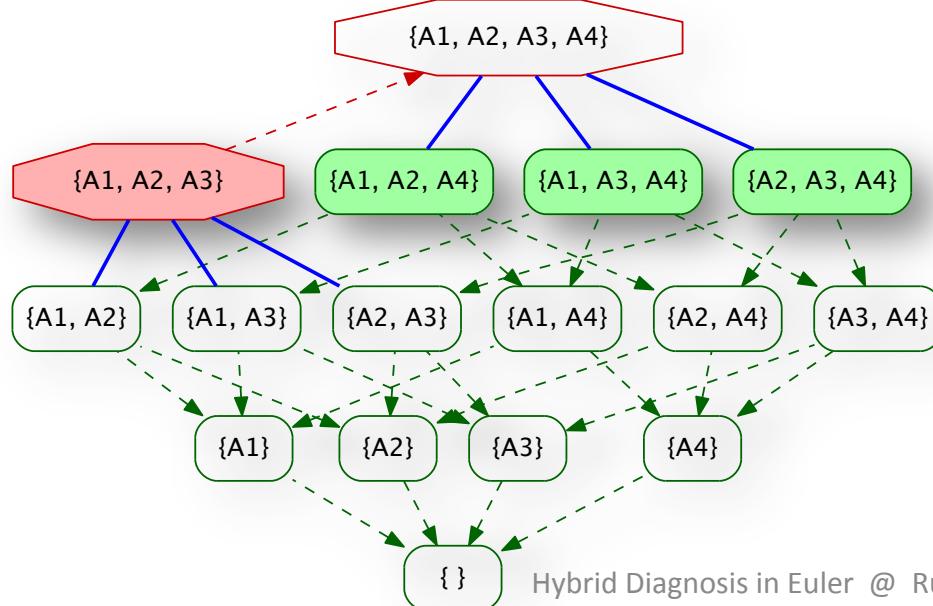
# Possible Outcome: Inconsistency!



**Inconsistent!**

→ Diagnosis (Reiter)

= Black-Box  
Provenance

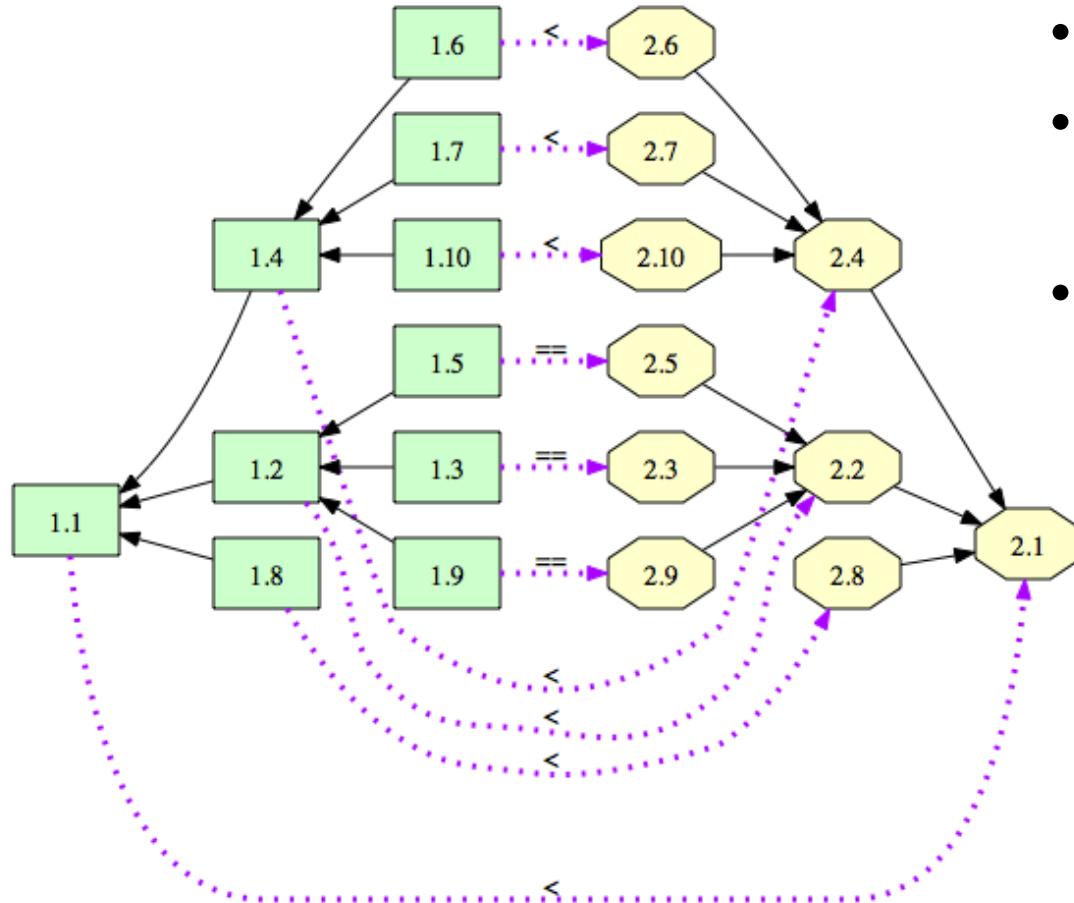


- Need to **debug** the input articulations → (black-box) diagnosis!
- Focus:
  - How do we **efficiently** compute the diagnostic lattice?
- Also:
  - How to **visualize**..

Nodes	
1	10
2	10
Edges	
isa_1	9
isa_2	9
Art.	10

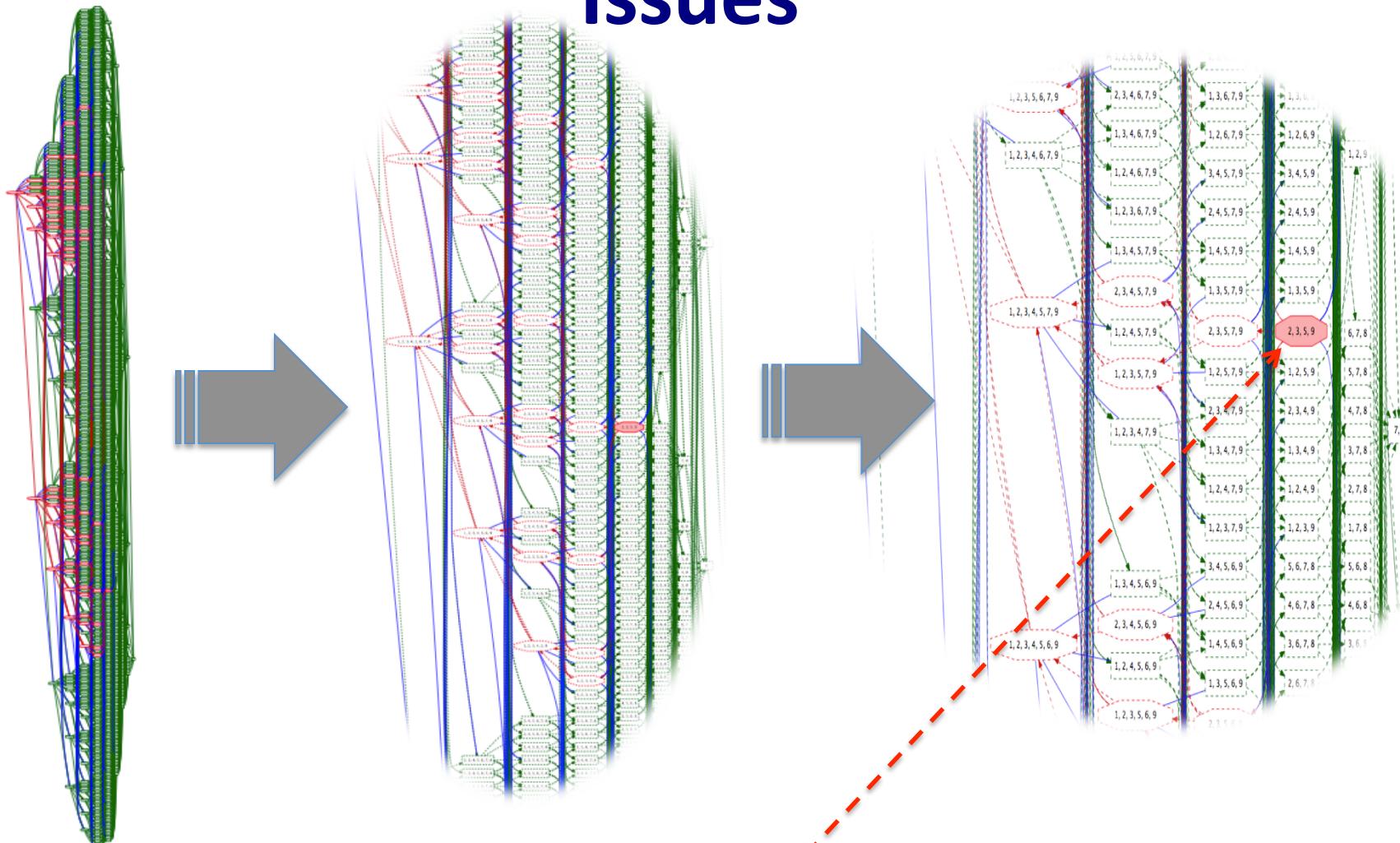
# Example Instance

(from synthetic benchmark suite)



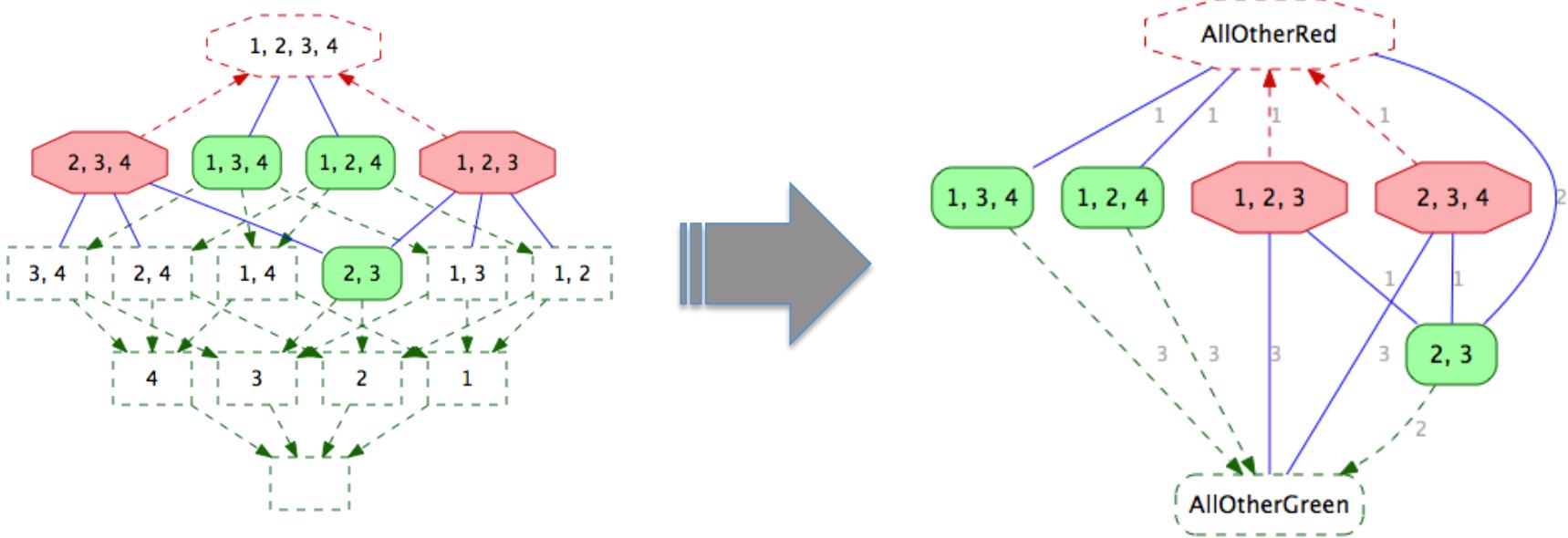
- Here:  $N = 10$  taxa in  $T_1, T_2$
- Euler/X finds:  
**inconsistent!**
- → **diagnostic lattice** of  $2^{10}$   
= **1024** nodes
  - Find **minimal inconsistent subset (MIS)**
  - maximal consistent subset (**MCS**) ..
  - **show to user!**

# Visualizing Diagnoses: Scalability Issues



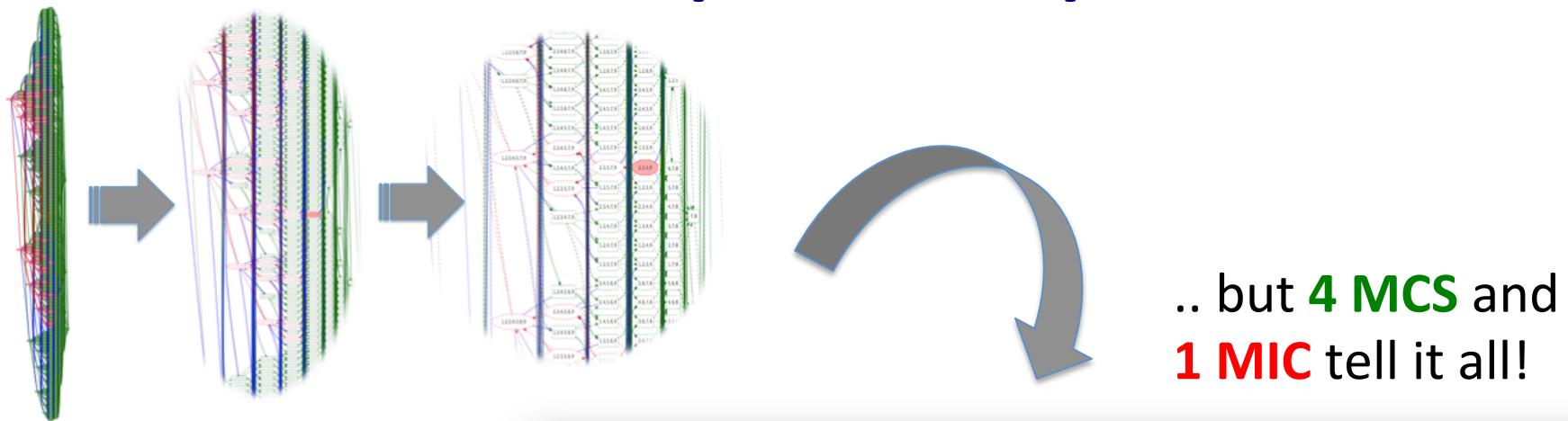
$N = 10$  articulations  $\rightarrow 2^{10} = 1024$  node diagnostic lattice (... ouch!)  
... but only one MIS ...

# Better Idea: Just show MIS, MCS

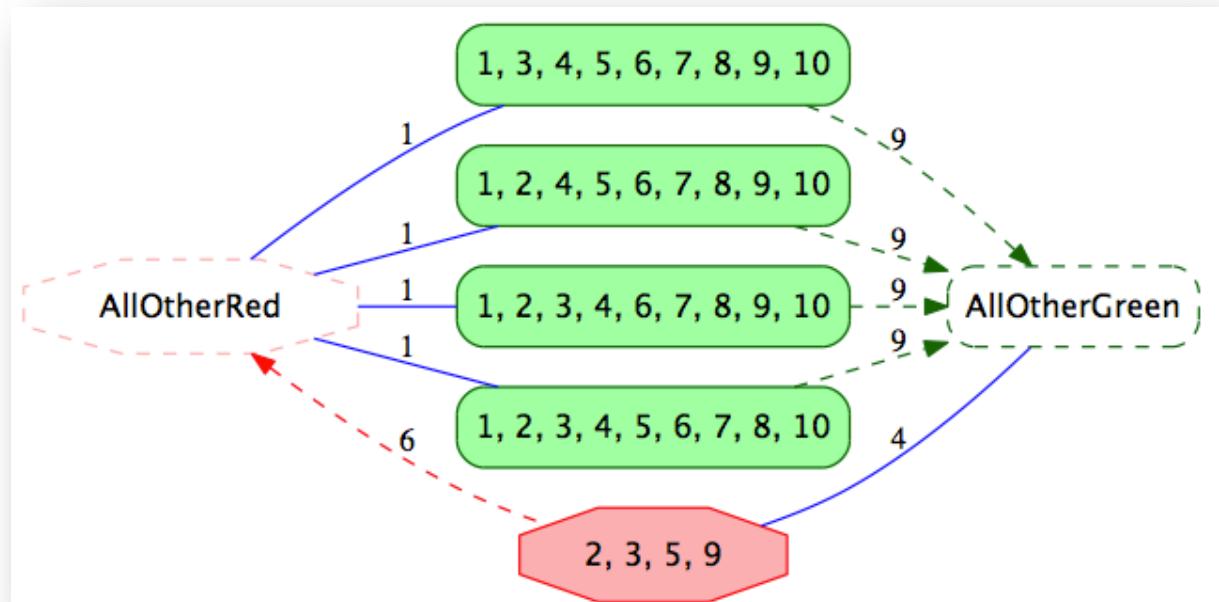


$N = 4$  articulations  $\rightarrow 2^4 = 16$  node diagnostic lattice,  
but 3 **MCS** and 2 **MIS** are enough!

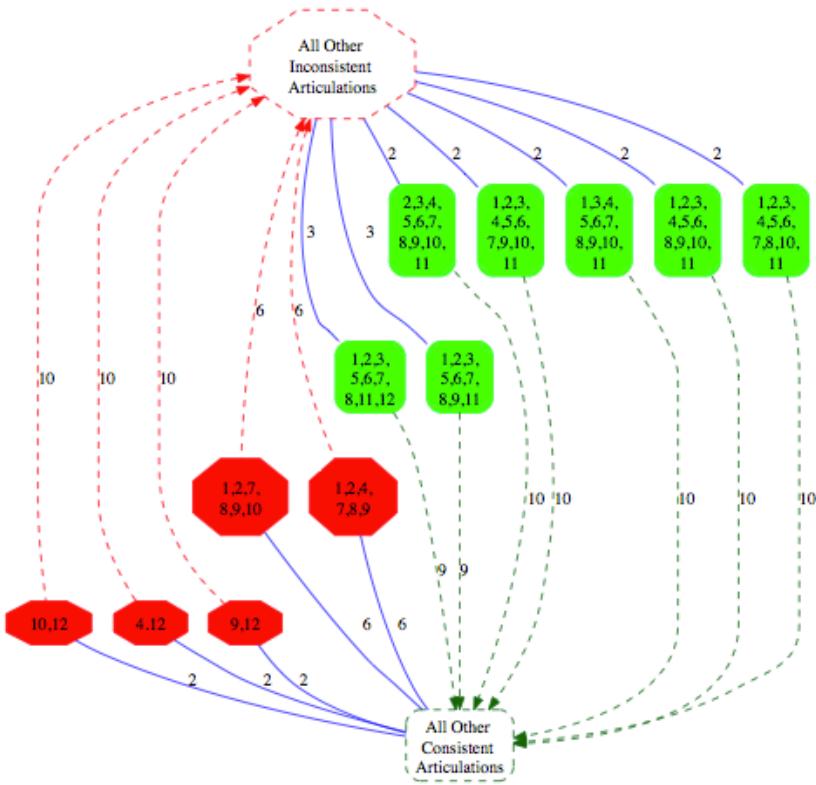
# Visualizing Diagnoses: Focusing on MIS (and MCS)



1024 node lattice



# Visualizing Diagnoses

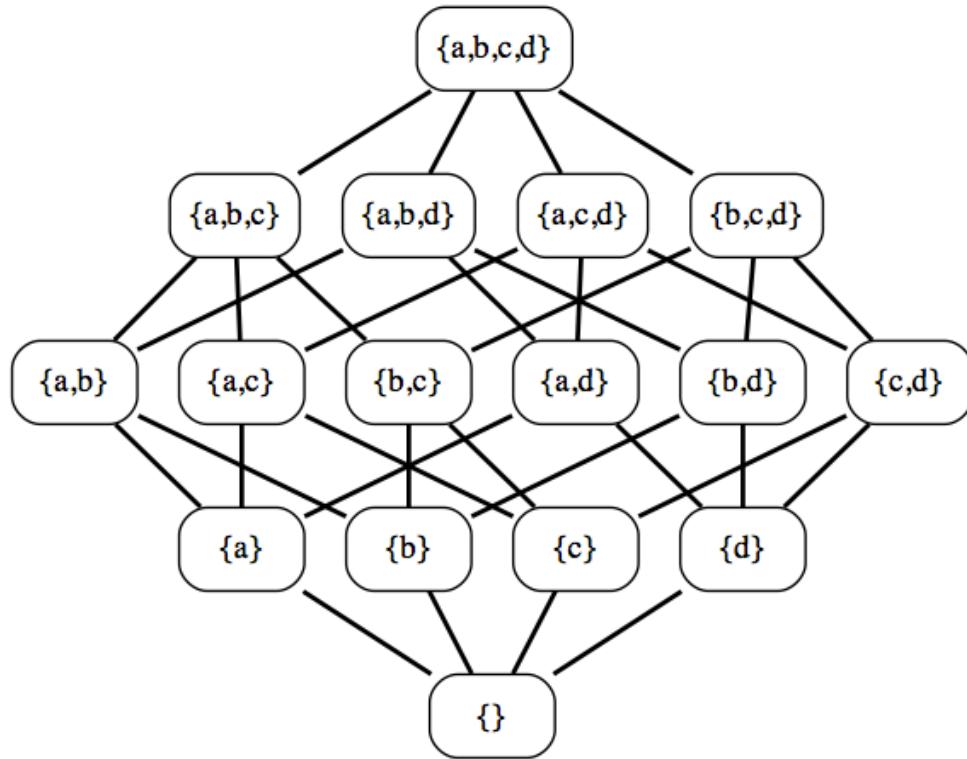


**Fig. 7.**  $MIS$  (Octagon) and  $MCS$  (Rounded Box) for Example 2. All other non-minimal inconsistent subsets and non-maximal consistent subsets are collapsed as “clouds”, the labels of edges show the path length from  $MIS/MCS$  to the top/bottom of the lattice.

Example from  
paper:  $N=12 \rightarrow$   
4096 nodes

.. but 7 MCS and  
**5 MIC** tell it all!

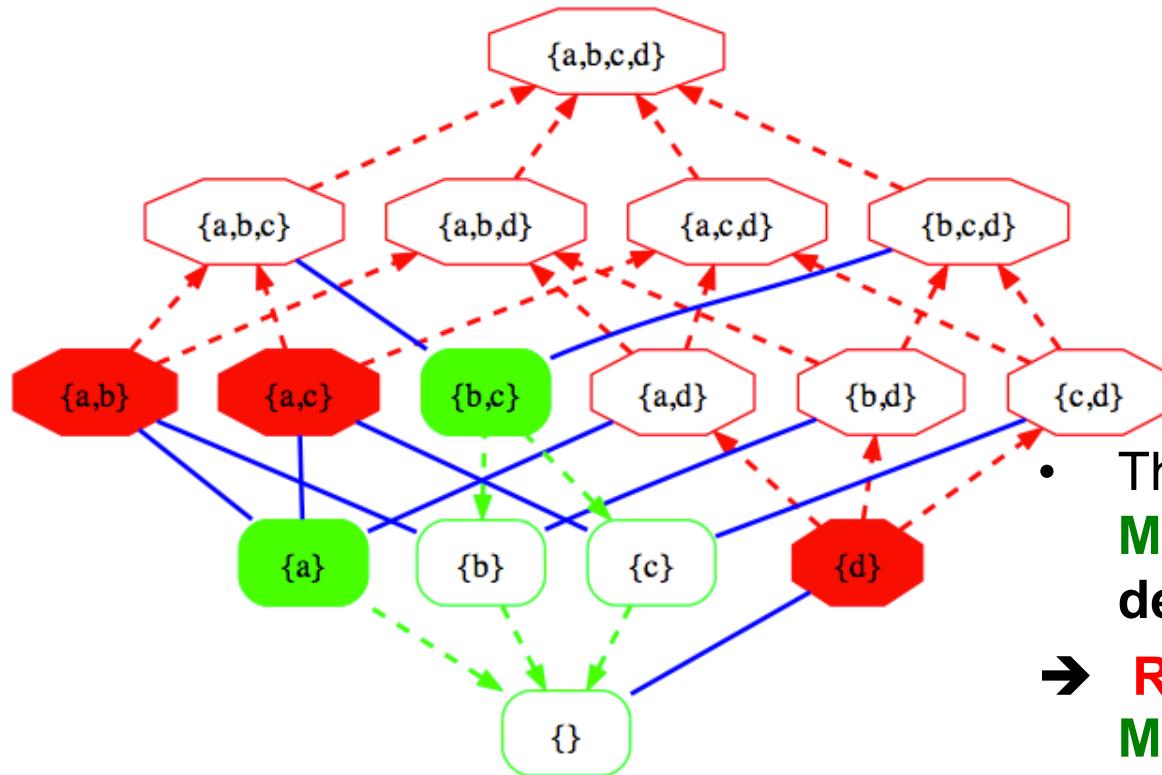
# Black-Box Inconsistency Analysis (Diagnostic Lattice)



*What happens if you can't have all (here: 4) articulations together?*

- Then:
  - Repair: find & revise **minimal inconsistent** subsets (**Min-Incons**)
  - Expand: find **maximal consistent** subsets (**Max-Cons**) & revise *outs*

# Inconsistency Analysis (Diagnostic Lattice)



- The **Min-Incons (MIS)** and **Max-Cons (MCS)** sets determine all others  
→ **Repair MIS** and/or **Expand MCS**

- **Black-box Analysis** (Hitting Set algo.) yields a **Diagnosis** (lattice)
  - for  $n=4$  articulations, there are 168 possible diagnoses
  - depending on expected “red/green areas” → explore space differently
- $|\text{articulations}| = n \rightarrow$ 
$$|\text{possible diagnoses}| = |\text{monotonic Boolean functions}|$$
$$= \text{Dedekind Number } (n): 2, 3, 6, 20, \mathbf{168}, 7581, 7828354, \dots$$

# Improving Diagnosis

- Reiter's "black-box" (model-based) diagnosis helps debug the articulations
- Limited scalability (inherent complexity)
- But every bit helps:
  - Hitting Set Algorithm ("logarithmic extraction")
- Our idea:
  - Exploit "white-box" reasoning information
  - **RULES** to the rescue

# Key Idea: exploit white-box info

- We use Answer Set Programming (ASP) to solve Taxonomy Alignment Problem (TAP)
- Inconsistency = “False” is derived in the head:  
**False :- <denial of integrity constraint>**
- Apply **provenance trick** from databases ☺
  - What articulations contribute to a derivation of “False” ?
  - **Eliminate those that don’t!**
  - ➔ an example of **reusing** inferences **across** separate black-box tests!

# The Provenance “Trick”

$$H_1(\bar{Y}, r_1 \otimes (P_1 \otimes \dots \otimes P_n)) :- B_1(\bar{X}_1, P_1), B_2(\bar{X}_2, P_2), \dots, B_n(\bar{X}_n, P_n).$$

For any constraint rule without head predicate (i.e., **false** is the head):

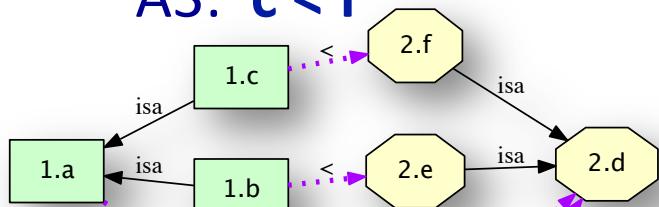
$$r_2 : \text{false} :- B_1(\bar{X}_1), B_2(\bar{X}_2), \dots, B_n(\bar{X}_n).$$

We rewrite it to a constraint with head predicate **NOK** where  $P_i$  is the provenance of  $B_i(\bar{X}_i)$  for  $1 \leq i \leq n$  and **NOK** stands for “Not OK”, i.e. inconsistency:

$$\text{NOK}(r_2 \otimes (P_1 \otimes \dots \otimes P_n)) :- B_1(\bar{X}_1, P_1), B_2(\bar{X}_2, P_2), \dots, B_n(\bar{X}_n, P_n).$$

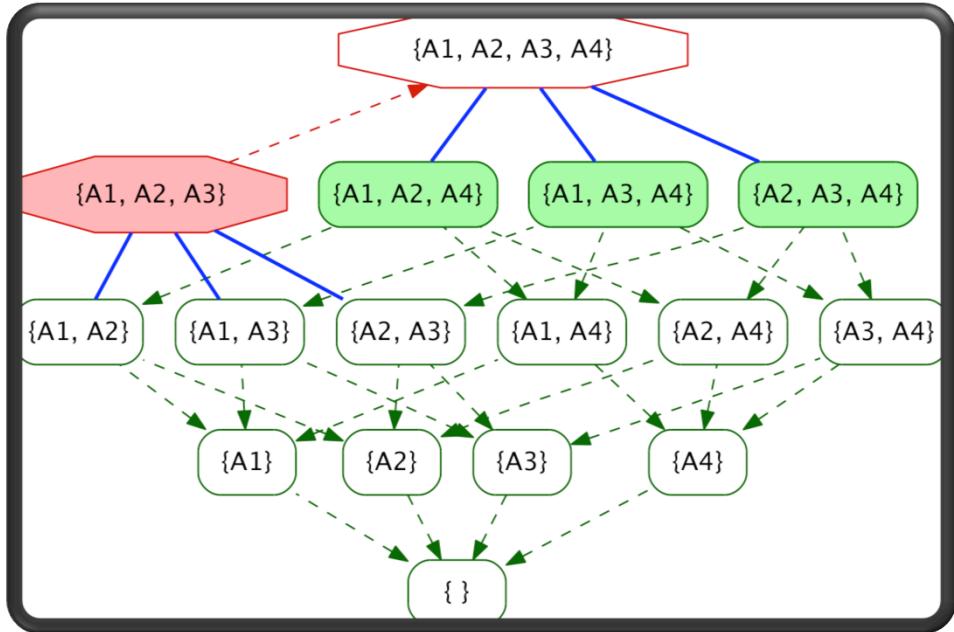
# Hybrid Provenance

A3:  $c < f$



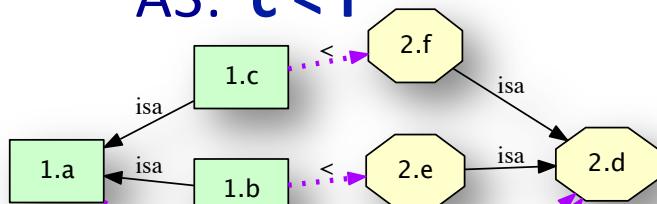
**Input Alignment**

*Black-box Provenance*



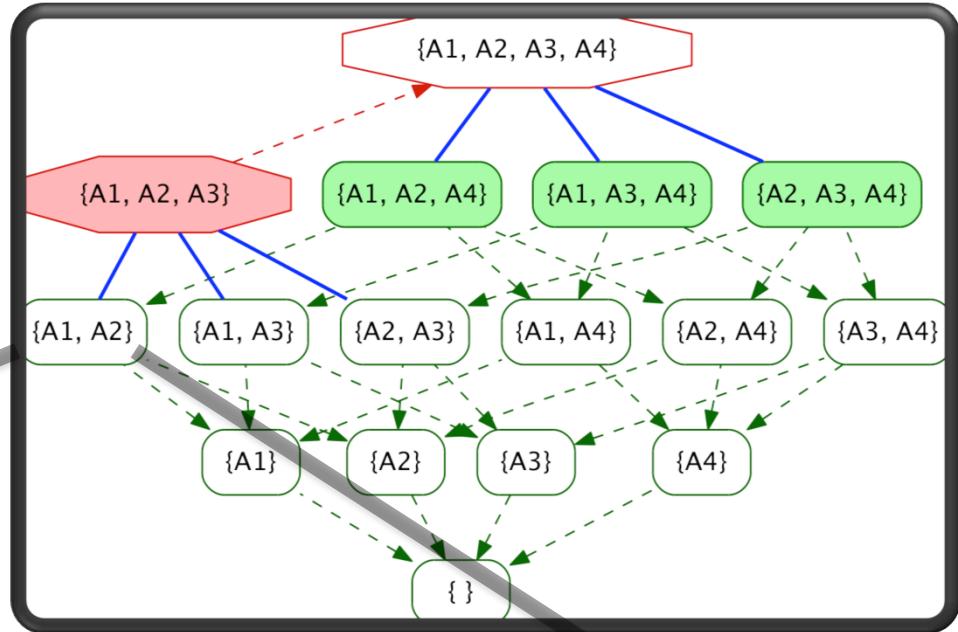
# Hybrid Provenance

A3:  $c < f$



**Input Alignment**

*Black-box Provenance*



r7:  $d = e \cup f$

A1:  $a = d$

$A1 + A2 + \dots \Rightarrow f < c$

$a = e \cup f$

r3:  $a = b \cup c$

r4:  $b \cap c = \emptyset$

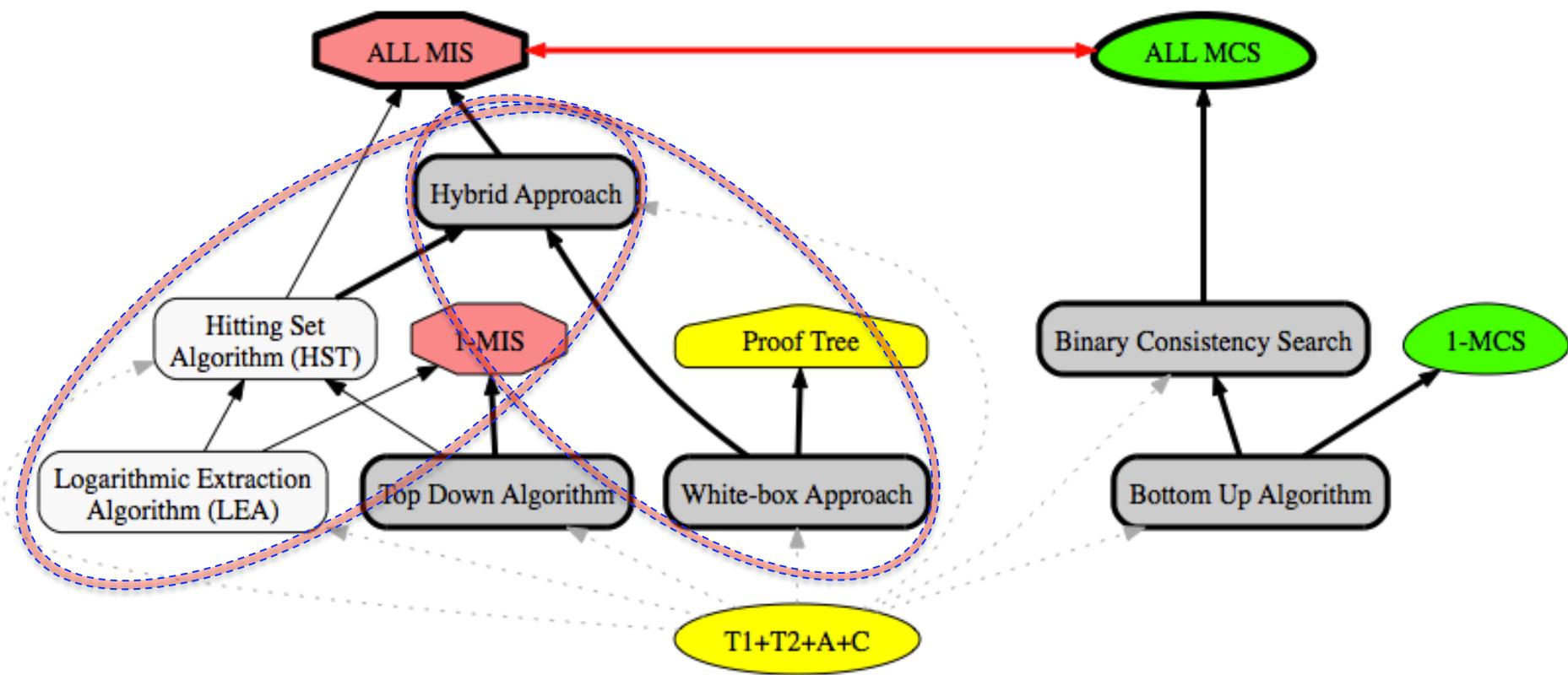
r8:  $e \cap f = \emptyset$

A2:  $b < e$

$f < c$

*White-box  
Provenance*

# The Hybrid Approach



# Hybrid Approach

## Algorithm 3 White-Box Approach

**Input:** System description  $SD$ , a set of constraints  $C$

**Output:** All diagnosis proof trees

**ComputeAllProofTrees( $SD, C$ ):**

- 1: Encode  $SD$  and  $C$  in Datalog rules
- 2: Rewrite Datalog rules to ones with provenance
- 3: Run ASP reasoner to get boolean expressions for NOK
- 4: Construct diagnosis proof trees using the boolean expressions

## Algorithm 4 Hybrid Approach

**Input:** System description  $SD$ , a set of constraints  $C$

**Output:** All diagnoses ( $MIS$ )

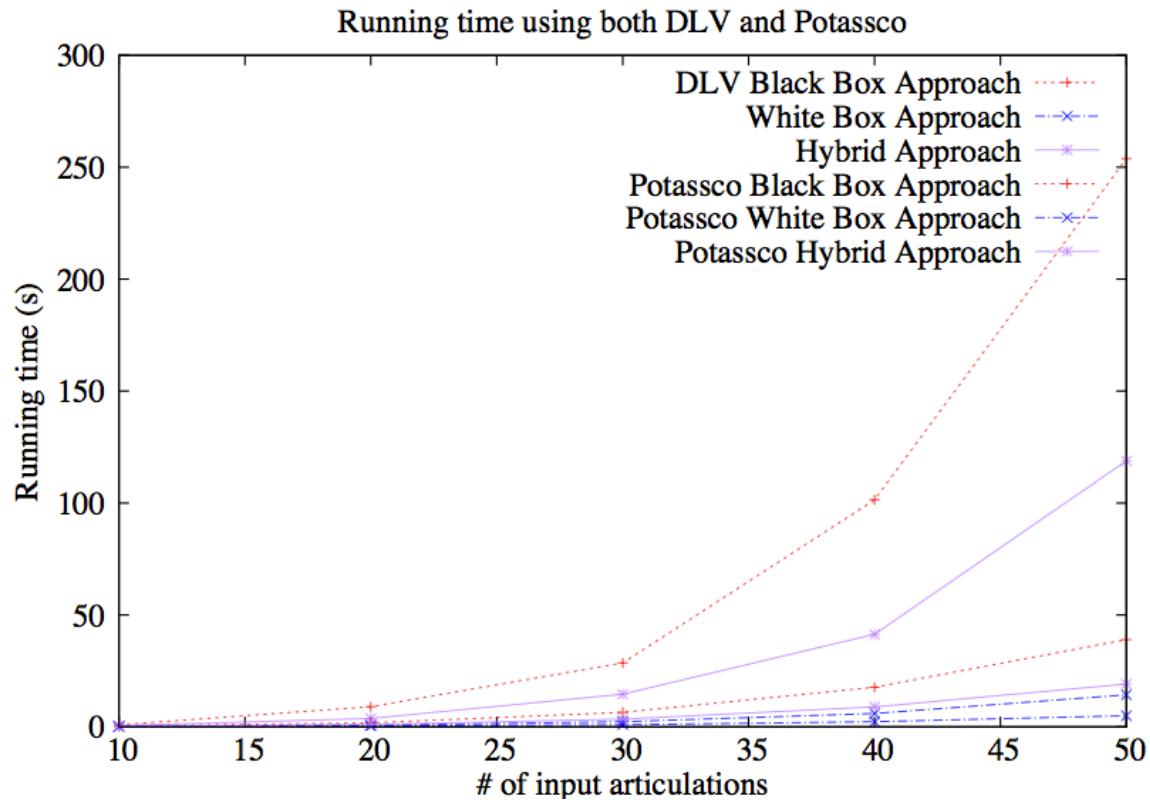
**ComputeAllMISHybrid( $SD, C$ ):**

- 1:  $Ts \leftarrow \text{ComputeAllProofTrees}(SD, C)$
- 2:  $C' \leftarrow$  set of leaf nodes of the proof trees  $Ts$
- 3: **return**  $\text{ComputeAllMIS}(SD, C')$

What articulations  
contribute to some  
inconsistency?

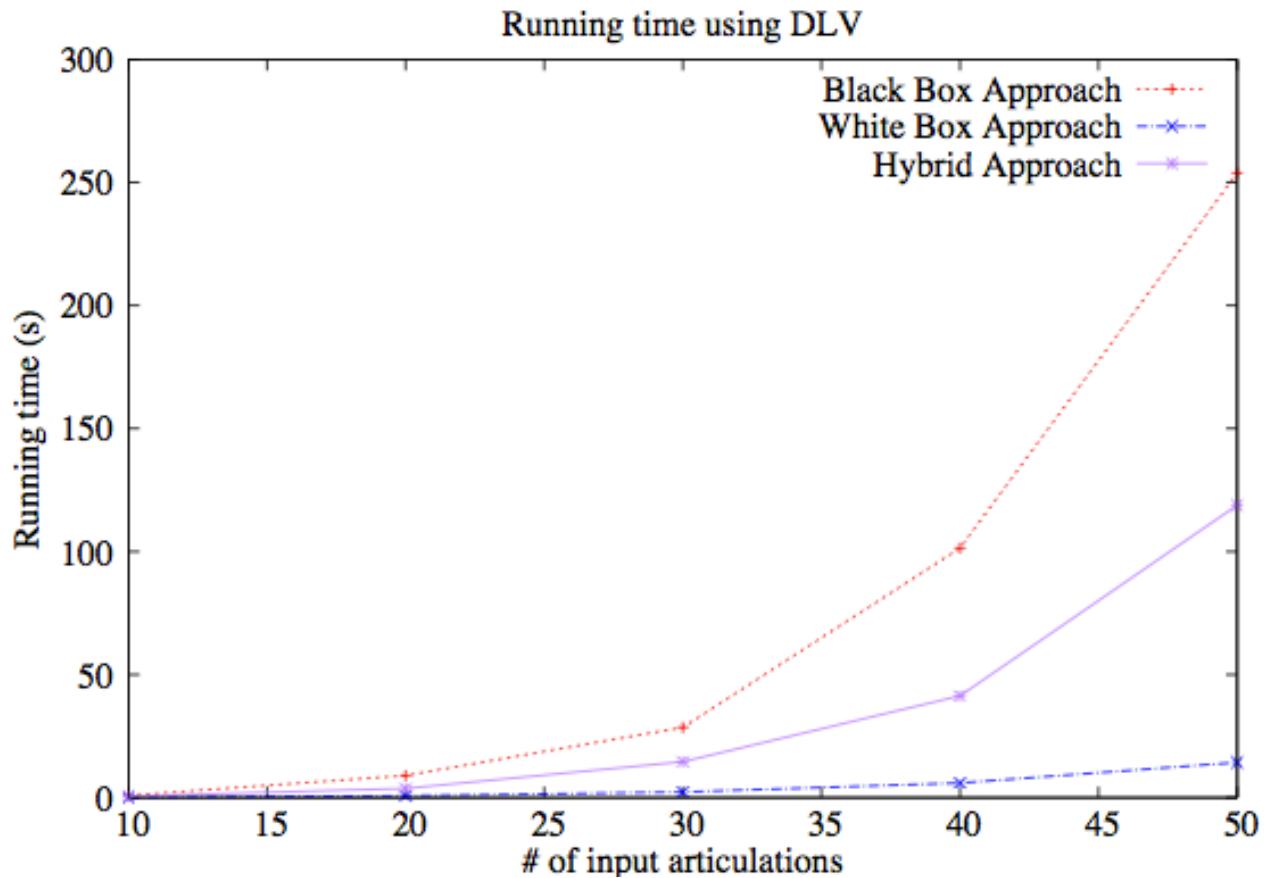
Good old black-box  
(HST)

# Benchmark Results



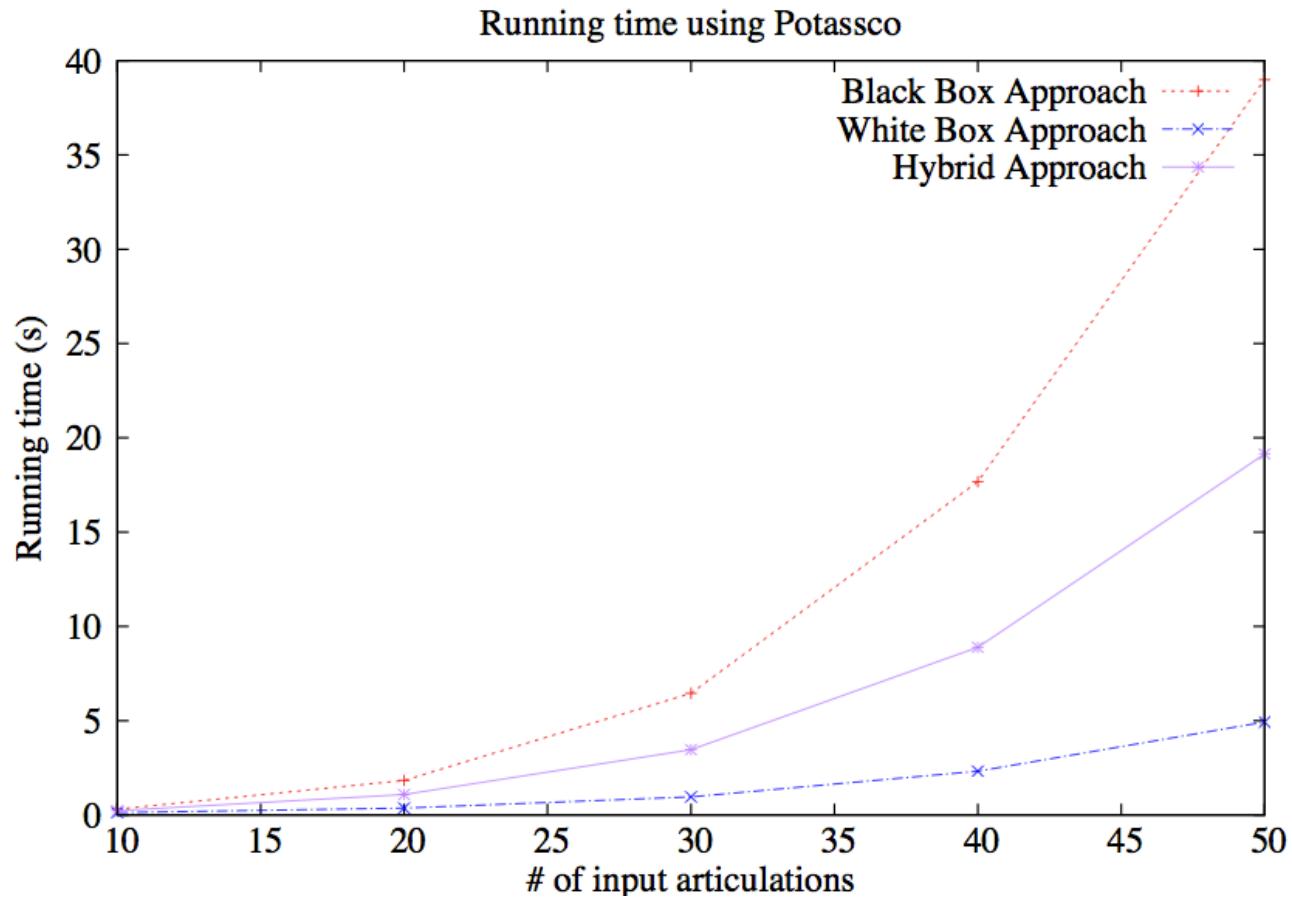
- White-box < Hybrid < Black-box (runtimes)
- Note: white-box does **not** give you a diagnosis
- Potassco < DLV

# Benchmark DLV



- White-box < Hybrid < Black-box (runtimes)
- Potassco < DLV

# Benchmark Clingo



- White-box < Hybrid < Black-box (runtimes)
- Potassco < DLV

# Conclusions

- ASP rules can be used to efficiently solve real-world taxonomy reasoning problems
- Reiter's diagnosis useful to debug inconsistent alignments
- **Adding a “white-box” provenance approach speeds up state-of-the-art HST algorithm by eliminating independent articulations**
- Future work:
  - Further improvements, including **parallelism**:
    - Trade-off with sharing inferences across parallel instances