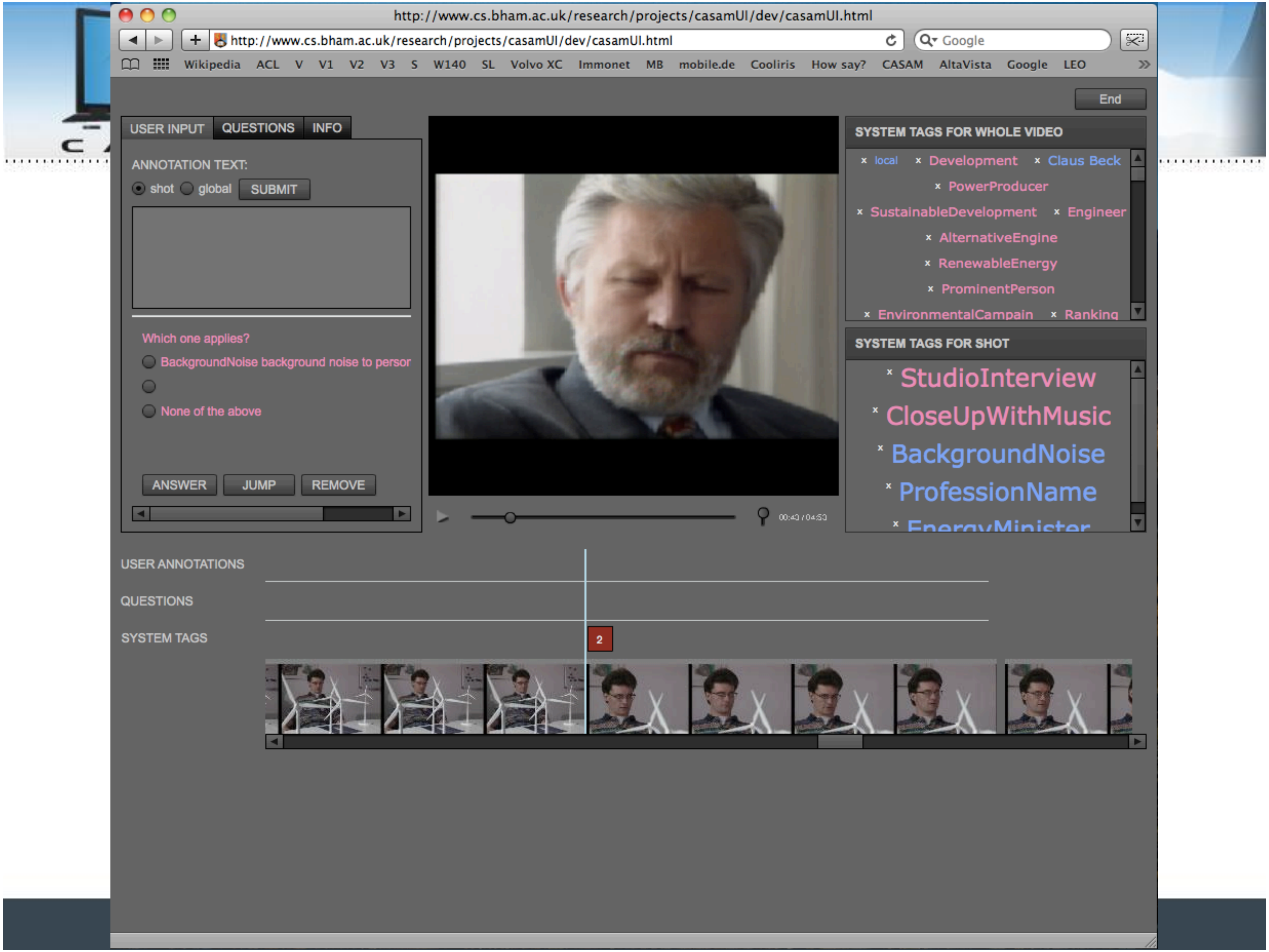
CASAM Schedule

Tasks

- Task 3.1: Optimized reasoning engine for probabilistic first-order structures (Lead TUHH)
  - New approach developed (Paper presented by Oliver at UniDL'10)
- Task 3.2: System supporting probabilistic abduction as a reasoning service (Lead TUHH)
  - Anahita presents paper at RR 2010
  - Michael's presentation
- Task 3.4: Meta-level reasoning component (Lead TUHH)
  - Query generation integrated into second prototype (Michael's pres)
  - See upcoming deliverable D3.4
- Task 6.2: MM Ontology: MESH ontology



http://www.cs.bham.ac.uk/research/projects/casamUI/dev/casamUI.html
Google
Wikipedia ACL V V1 V2 V3 S W140 SL Volvo XC Immonet MB mobile.de Cooliris How say? CASAM AltaVista Google LEO
End
USER INPUT
QUESTIONS
INFO
ANNOTATION TEXT:
shot
global
SUBMIT
Which one applies?
BackgroundNoise background noise to persor
None of the above
ANSWER
JUMP
REMOVE
SYSTEM TAGS FOR WHOLE VIDEO
local
Development
Claus Beck
PowerProducer
SustainableDevelopment
Engineer
AlternativeEngine
RenewableEnergy
ProminentPerson
EnvironmentalCampain
Ranking
SYSTEM TAGS FOR SHOT
StudioInterview
CloseUpWithMusic
BackgroundNoise
ProfessionName
USER ANNOTATIONS
QUESTIONS
SYSTEM TAGS
2

New RMI Implementation

- Overview (I can skip slides on request)
  - Implemented architecture
  - Computation of queries
  - Optimization of abduction
  - Open issues
- CASAM Team @ STS / TUHH
  - Anahita Nafissi
  - Oliver Gries
  - Ralf Möller
  - Maurice Rosenfeld
  - Kamil Sokolski
  - Michael Wessel

What's new in a Nutshell...

- Agenda-based
  - manages RMI interpretations as small individual ABoxes

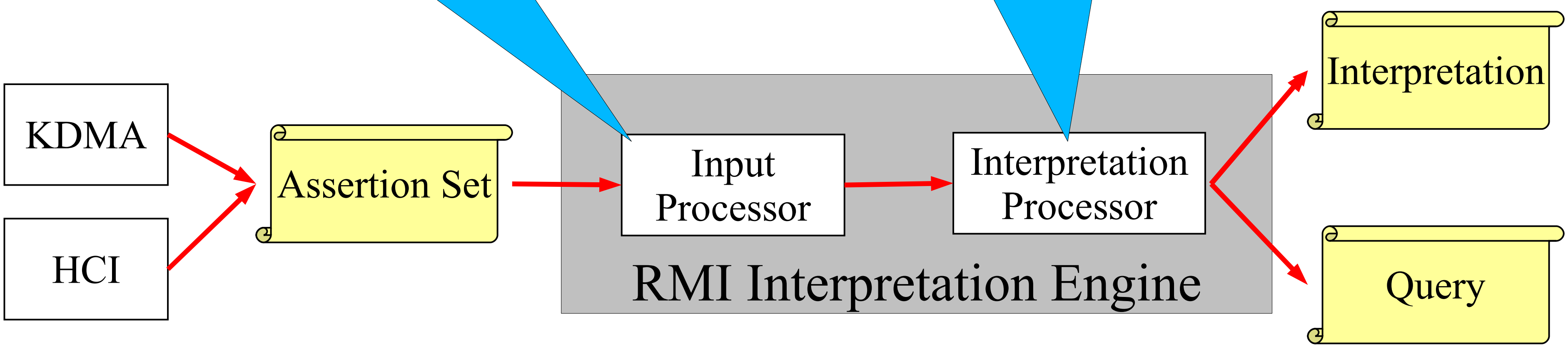
    + big „common part“ ABox CP (segments, EDO/MCO stuff, …)
  - incremental : **only reinterprets what needs to be reinterpreted**
    - uses only the relevant subset of CP (20% of CP) for Fiat rules
    - abduction performed on **subset** of CP + best interpretation

      → even „higher levels“ of interpretation possible
  - more control on interpretation process, by looking at the agenda (more information explicitly available) → **meta level reasoning**
- Queries computed for interpretations on agenda
- Lisp-based & multi-core ready
  - shares memory structures with RacerPro (no more OWL-in-out)

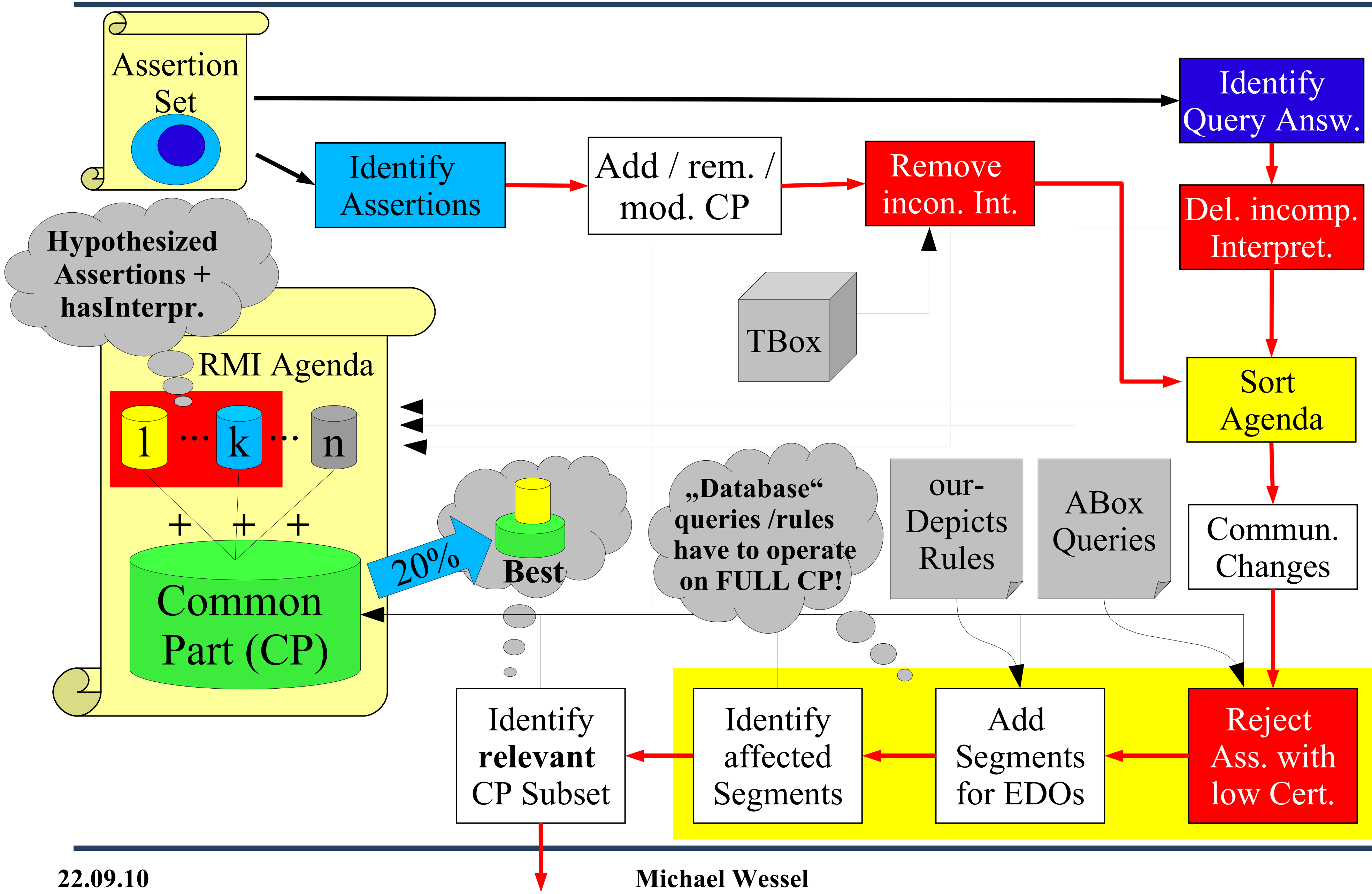
RMI Implementation of `receiveAssertions`

- Manage agenda (updates, query answers, ...)
- Abox augmentation
- Determine focus, compute relevant part of CP ABox

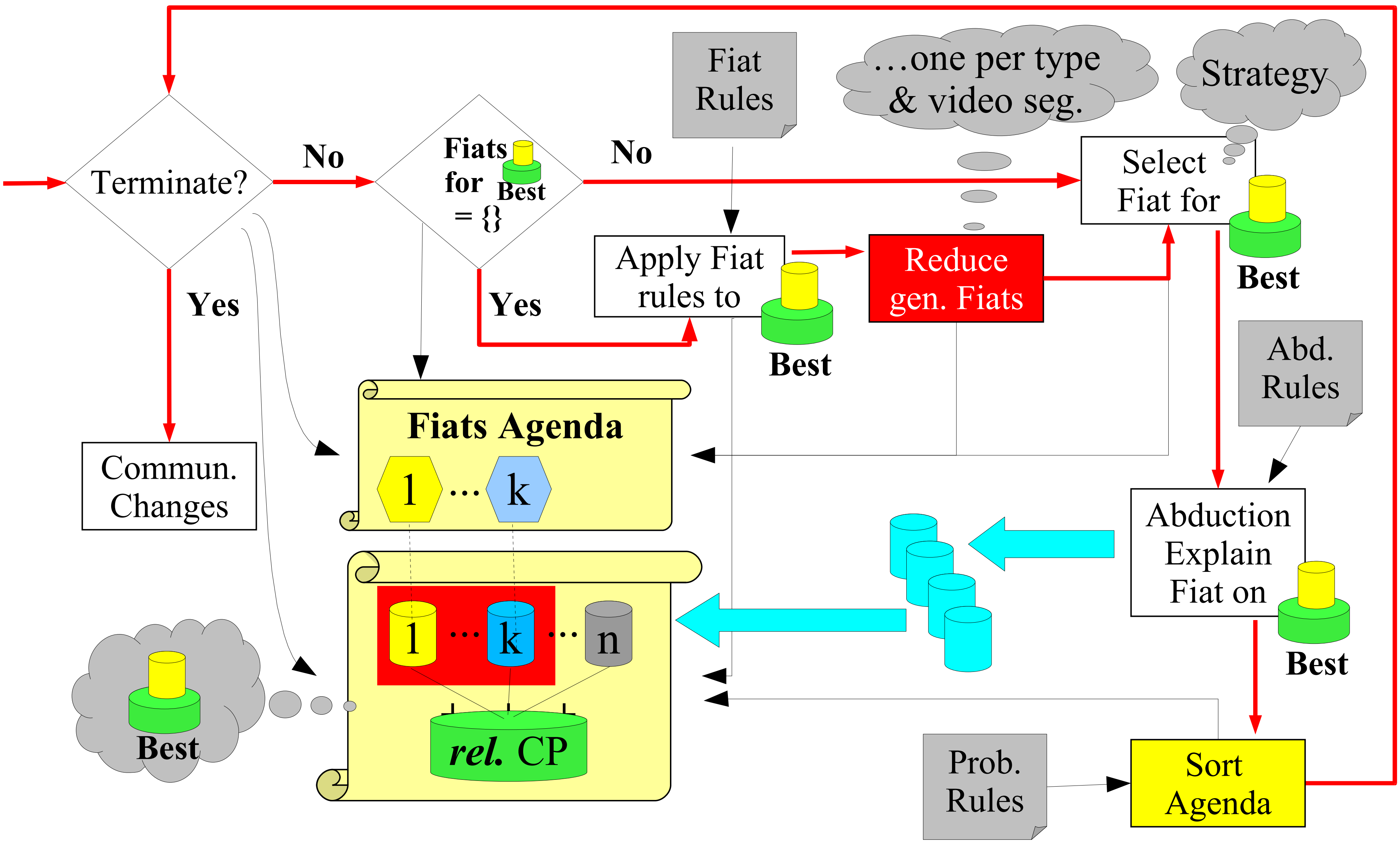
Perform the abduction in a loop until termination criterion met (max. # fiats, no more fiats, no probability increase, ...)



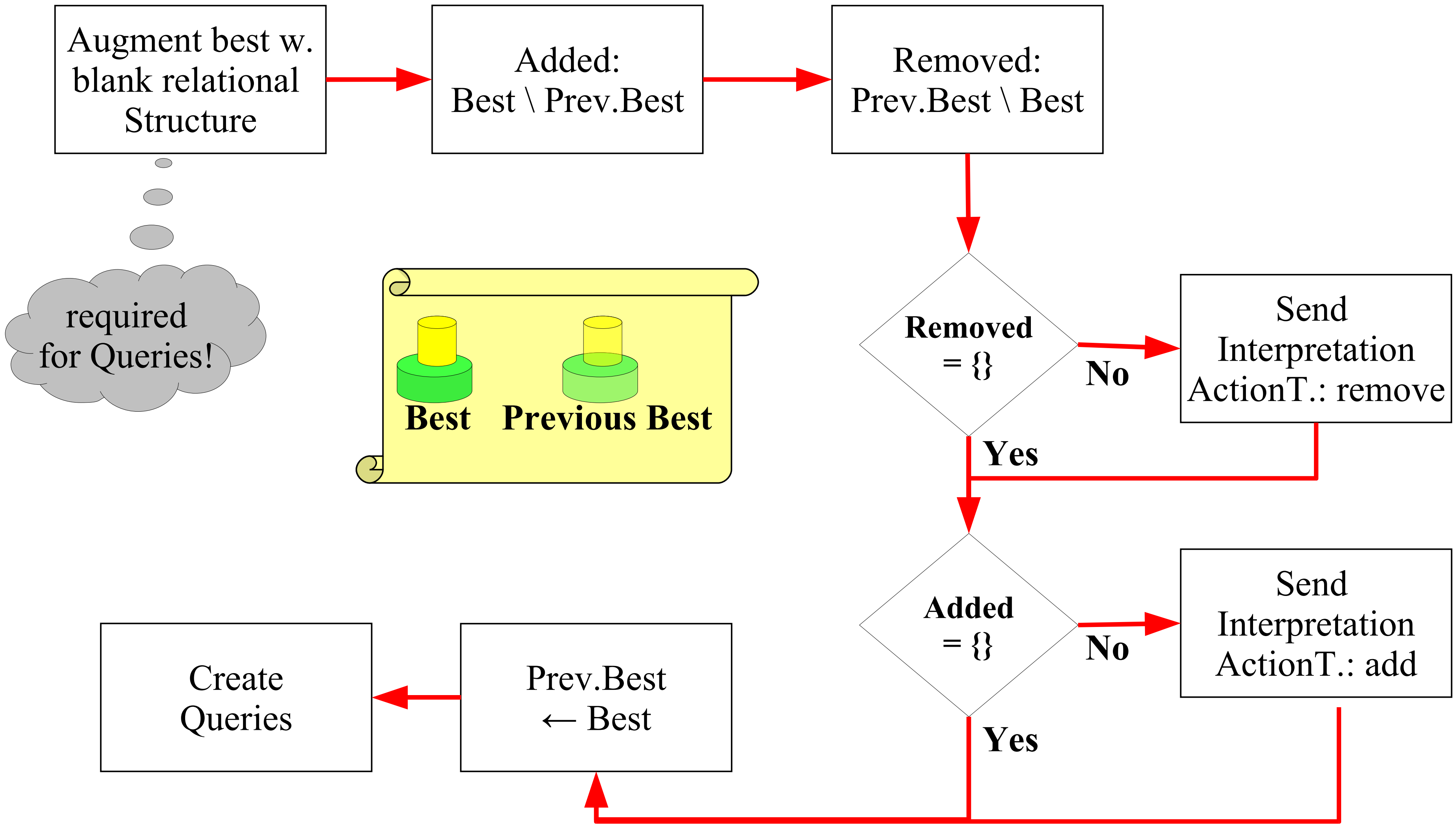
RMI Input Processor



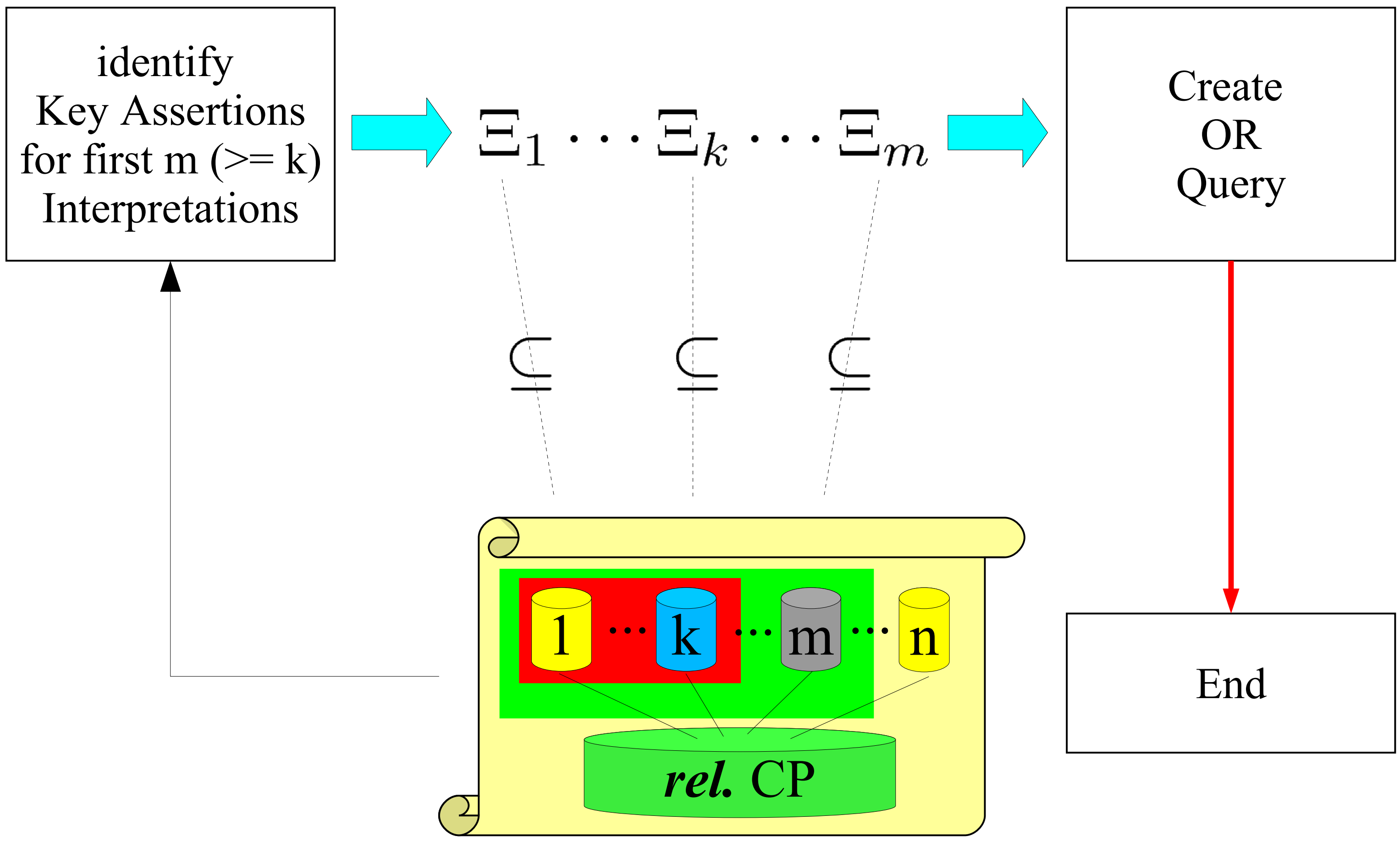
RMI Interpretation Processor



RMI Communicate Changes



RMI Create Queries



Computation of Queries

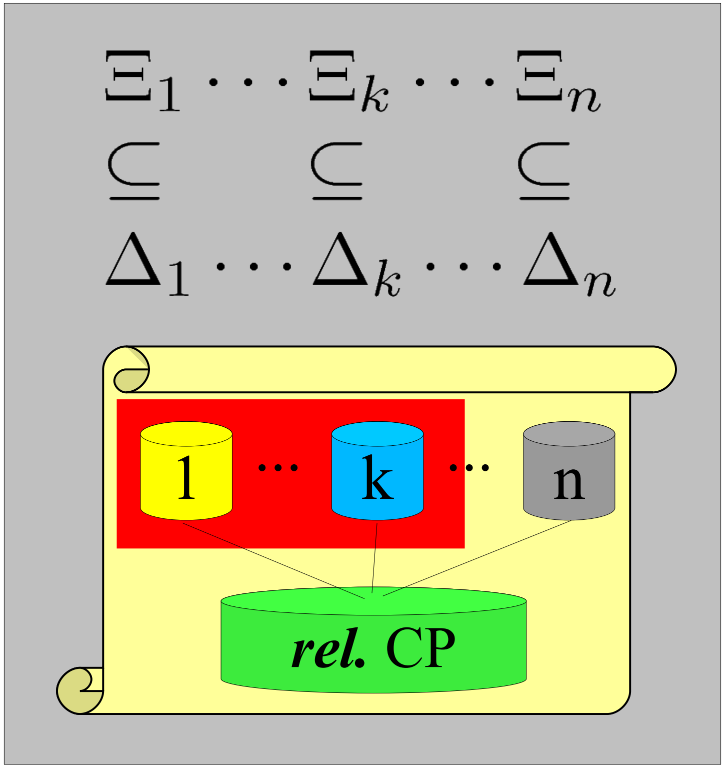
- Computation of characteristic („key“) assertions Ξ_i for $\Delta_i, 1 \leq i \leq n$
- Compute the „common differences“ by intersecting all differences to all other Δ_j

$$\Xi_i = \bigcap_{i \neq j, 1 \leq j \leq n} \Delta_i \setminus \Delta_j$$

- From each Ξ_i select an assertion (preferable an instance assertion)

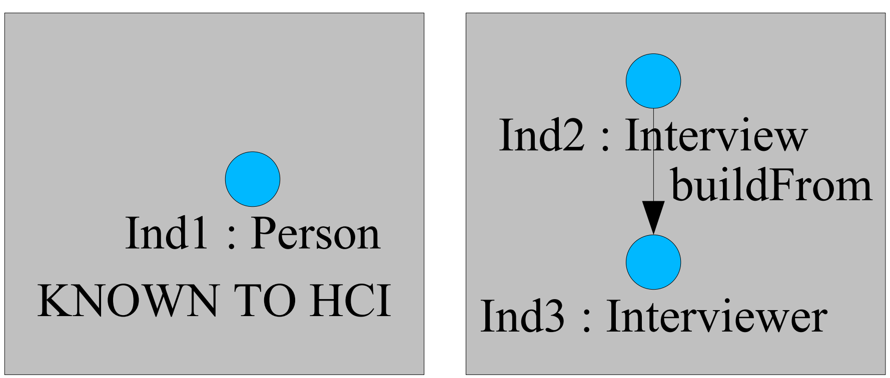
  → n disjuncts for OR query
  → simple score: 1 – 1 / *n*

- „\“ may be ABox difference, but...

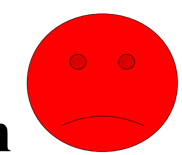


What is the blank relational structure and why is it required?

- Problem:
  - queries can only be formulated against the communicated „best“ interpretation: Δ_i
  - However, all but one query disjuncts come from $\Xi_j \subseteq \Delta_j$
  - the relational structures may be completely different
    - different hypothesized **RMI INDs**, different edges, etc.
- Example: how to communicate the difference between



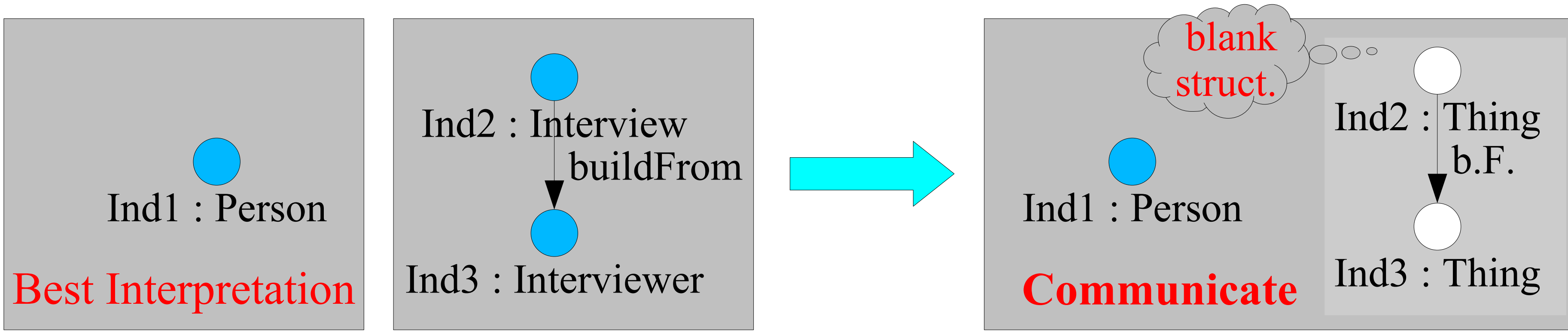
- HCI only knows **Ind1**!
- Q-Disjunct1: **Ind1 : Person**
- Q-Disjunct2: **Ind1 : Interview** ??
  Ind1 : Interviewer ??
- **Solution: avoid the problem in the first place!**

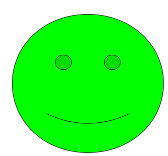


What is the blank relational structure and why is it required? (2)

- Instead of only sending the best interpretation, we also include the „blank relational structure“ of ALL other interpretations

→ relational structure and all hypothesized INDs known to HCI





- HCI knows **Ind1, Ind2, Ind3**!
- Q-Disjunct1: **Ind1 : Person**
- Q-Disjunct2: **Ind2 : Interview**
  [Ind3 : Interviewer]
  [(Ind2, Ind3) : b.F.]
- **No „new-ind mapping“ needed**

Augment best w. blank relational Structure

Abductive Query Answering

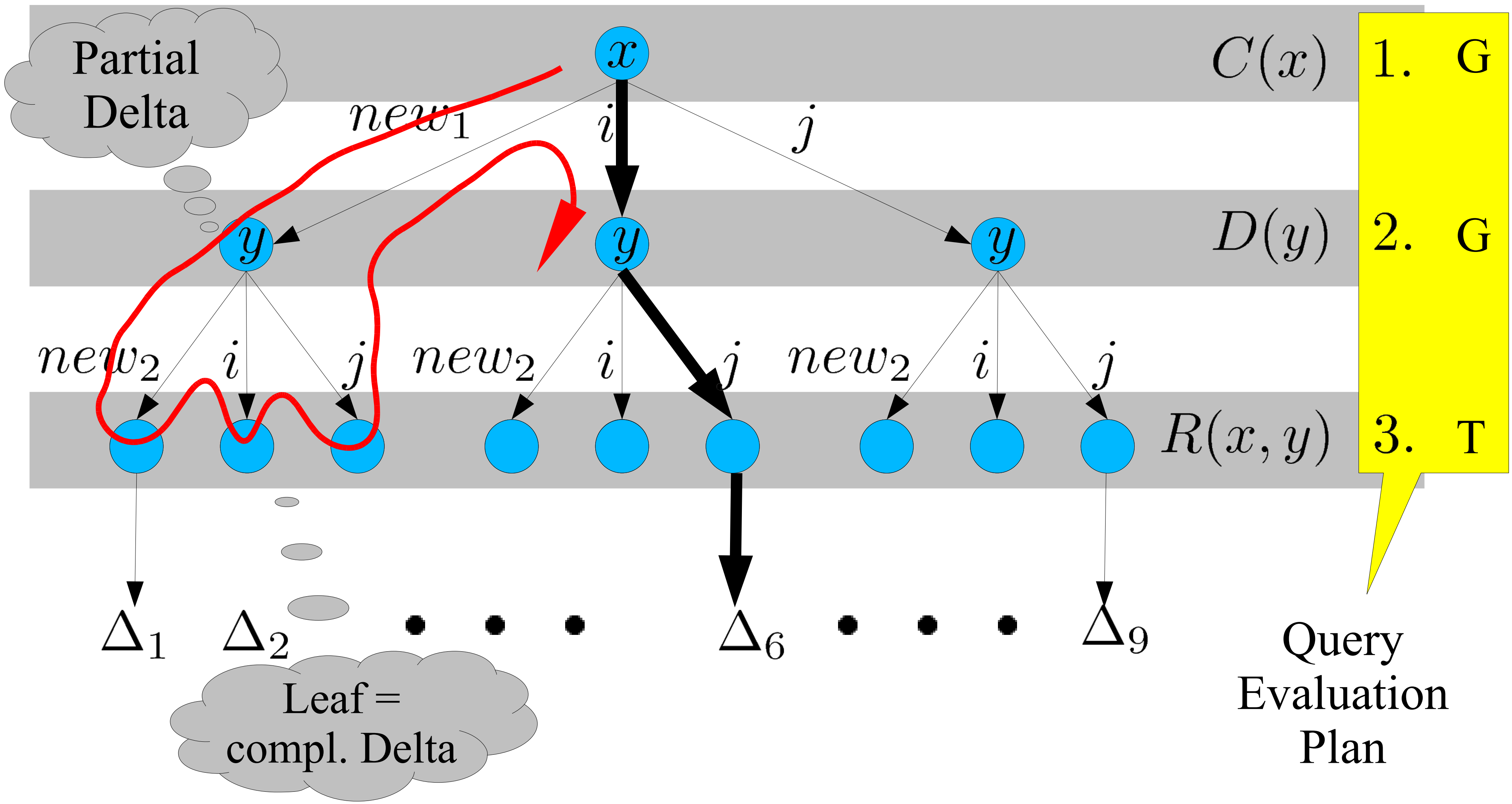
- Simple example
  - Query: $ans() \leftarrow C(x), D(y), R(x,y)$
  - Abox: $\{(i,j) : R, i : C\}$
  - **Preferred** solution (optimal, according to score defined below)

    $x \leftarrow i, y \rightarrow j :$
    $\Delta = \{j : D\}$
  - **Other** solution (plus 7 more, $3^2 = 9$), e.g.

    $x \leftarrow new1, y \leftarrow new_2 :$
    $\Delta = \{new_1 : C, new_2 : D, (new_1, new_2) : R\}$
- Exponential number of solutions has to be computed to find „the best“
  - **optimization idea:** early dynamic cutoff of search space based on score evaluation on partially computed explanations (deltas)

„Depth First“ Abductive Query Evaluation

$$\mathcal{A} = \{(i,j) : R, i : C\}$$



CASAM Preference Score

Very simple:

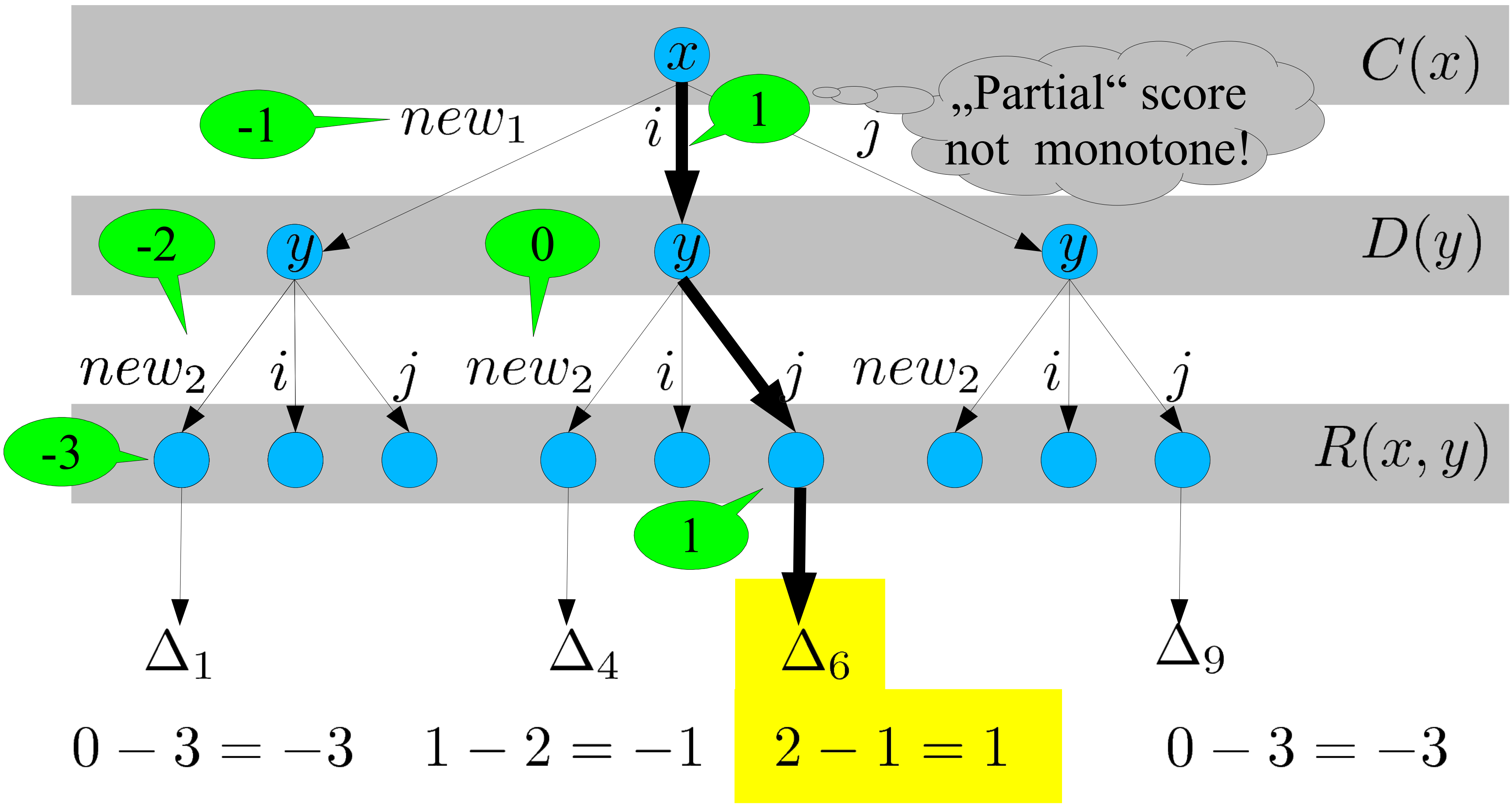
entailed Assertions minus hypothesized Assertions

$$\mathsf{score}(\Delta) =_{def} |\Delta^+| - |\Delta^-| \rightarrow \text{ maximize}$$

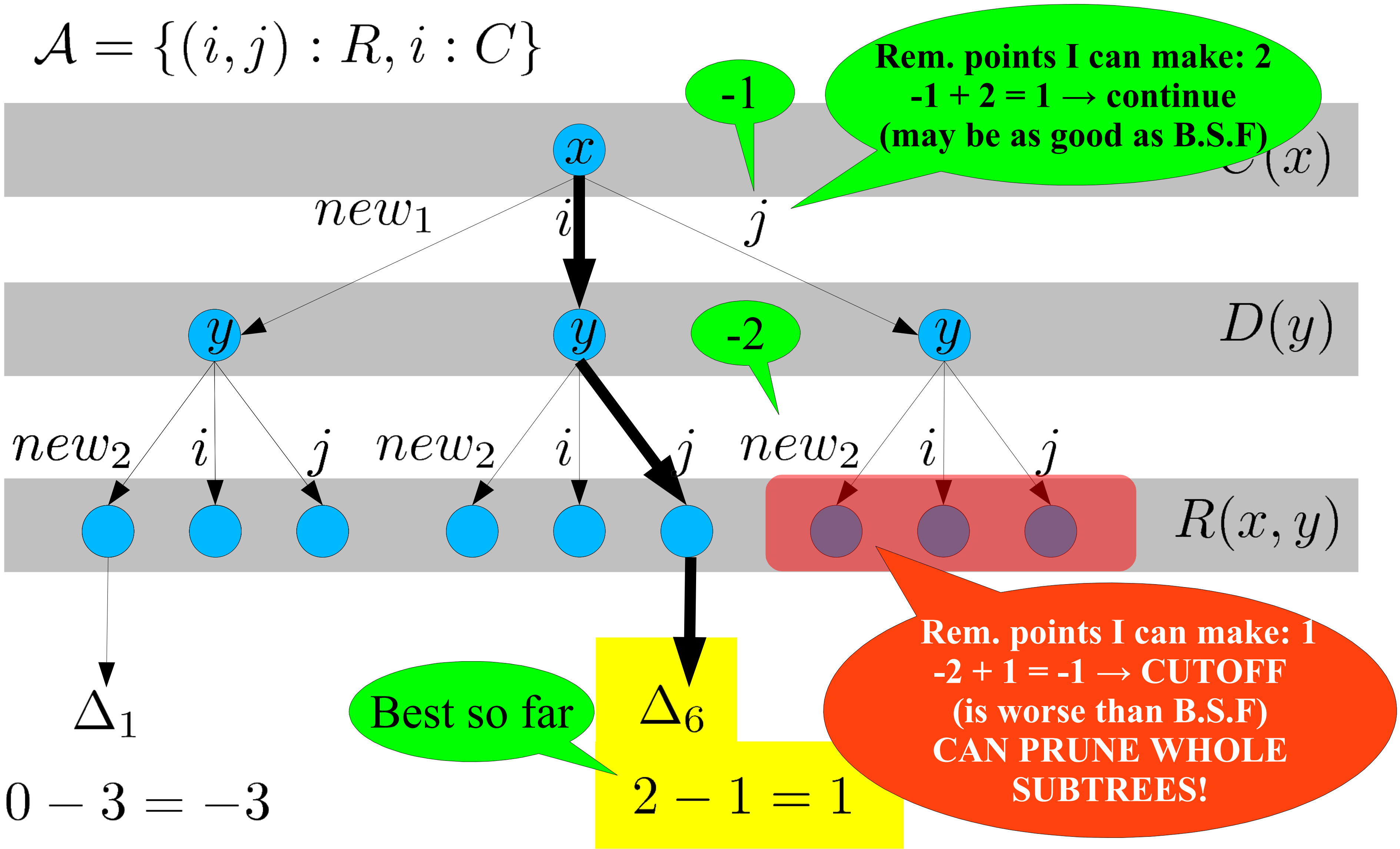
$$\Delta = \Delta^+ \cup \Delta^- \text{ (entailed, hypothesized)}$$

Illustrations of (Partial) Scores

$\mathcal{A} = \{(i,j) : R, i : C\}$ $\qquad |\Delta^+| - |\Delta^-| = score \rightarrow max.$



Score-Based Cutoff of Search Space



More formally...

$n = |\Delta^+| + |\Delta^-|$ (n const. for each rule body)

$\mathsf{score}(\Delta) =_{def} |\Delta^+| - |\Delta^-| \rightarrow$ maximize (not monotone)

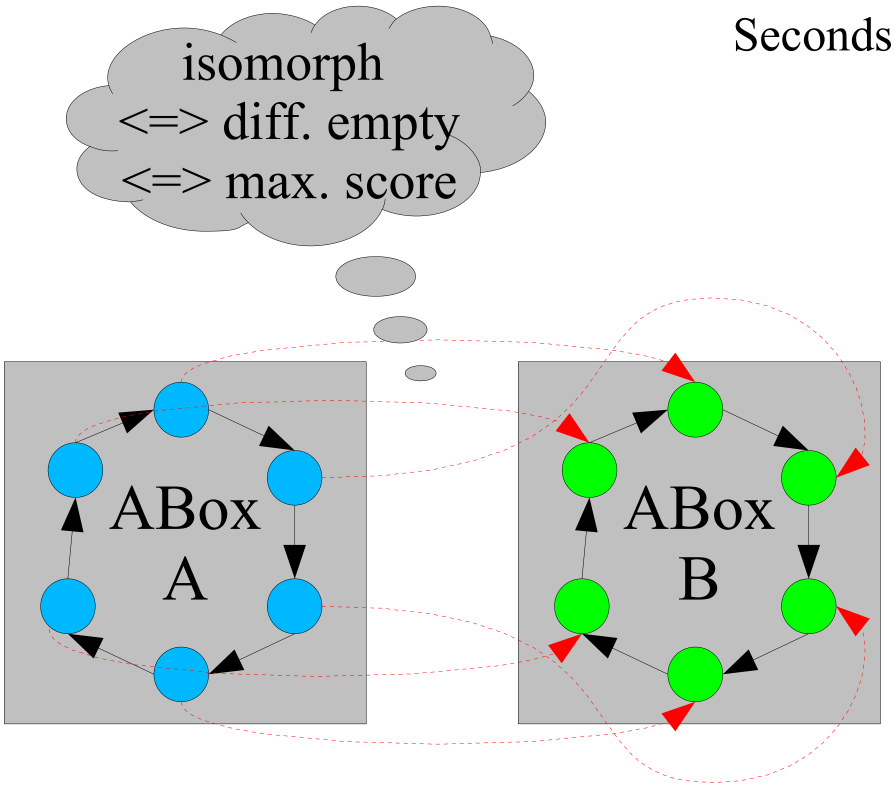
$n + \mathsf{score}(\Delta) = 2|\Delta^+|$

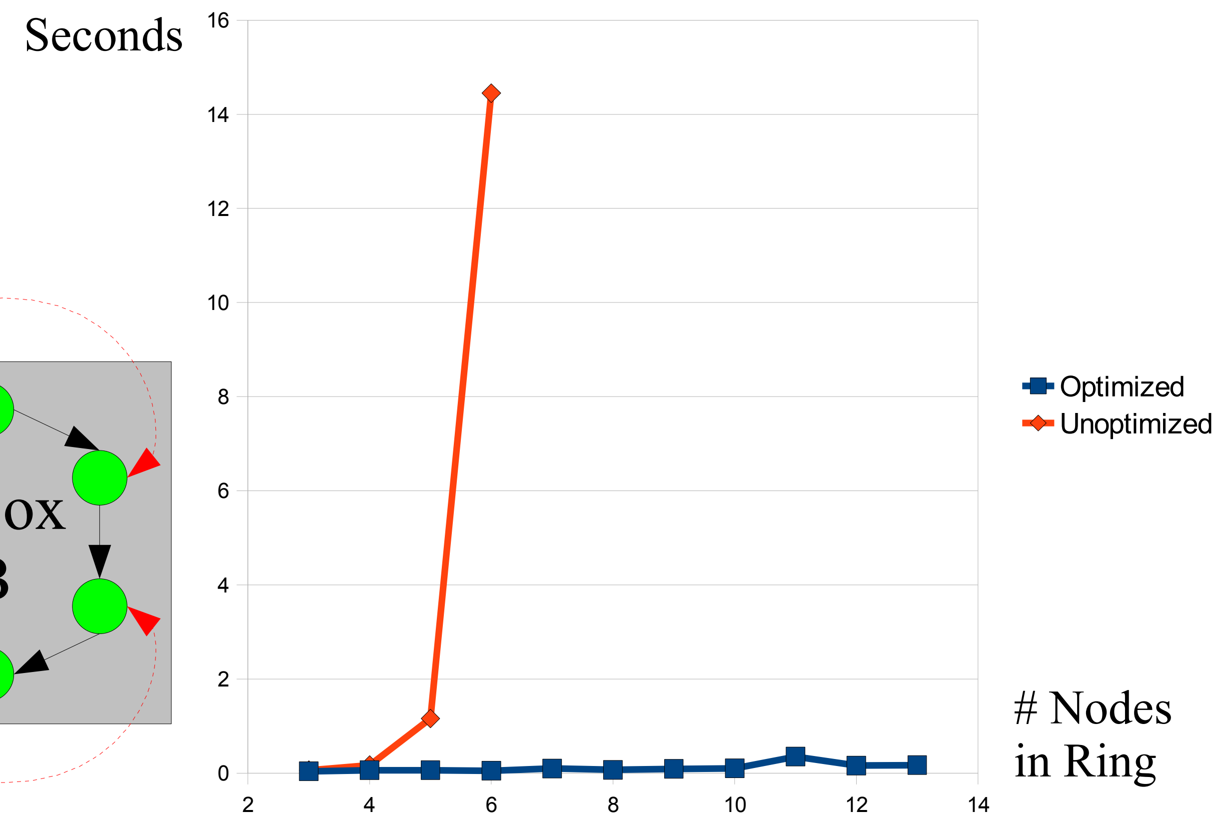
$\mathsf{score}(\Delta) = 2|\Delta^+| - n \rightarrow$ maximize (and monotone!)

- Let $\Delta_p \subseteq \Delta, m_p = n - |\Delta_p|$ (remaining conjuncts)
  - If $\mathsf{score}(\Delta_p) + (n - |\Delta_p|) < \mathsf{score}(\Delta_{best\_so\_far})$
    $\mathsf{score}(\Delta_{best\_so\_far}) - \mathsf{score}(\Delta_p) > (n - |\Delta_p|)$
    reject Δ_p

How Effective is this?

- Synthetic benchmark: finding graph isomorphisms (n nodes)
- Problem reductions:
  Graph Isomorphism → ABox Difference → Abduction





Appreciation of Complexity

- Some numbers
  - video 6, after bunch 3: 283 Fiats (new rule set)
    - potential quadratic number of Fiats (in terms of inds in the Abox)
    - after reduction „only one Fiat per type and shot“: 46 Fiats

Reduce gen. Fiats

  - „external complexity“ of interpretation loop
    - each Fiat may generate 2 to 3 explanations
    - branching will easily kill the system
  - „internal complexity“ of abduction (hidden in RacerPro)
    - in order to find these 2 to 3 best explanations PER FIAT, yet another exponential number of explanations has to be considered!
    - exponential in the number of indiviuals in the ABox

→ RMI handles **serious complex problems, more must be done for meta reasoning** (we stop after 30 Fiats per bunch)

Open Issues

- Reimplementation of probabilistic valuation and **Sort Agenda**
- React to removed / confirmed tags
- React to „negative“ query answers
  - only positive query answers considered so far
  - „shuffle“ the interpretations containing the answer assertions to the front of the agenda
- More specific Fiat generation rules
- Anytime / meta reasoning
  - reduce set of assertions if timeout occurs, etc.
  - some dumb strategies already implemented
- Q: do we really have to keep all interpretations on the agenda?