Scalable Reasoning: Cutting Ontologies Down to Size

Achille Fokoue, Aaron Kershenbaum, Li Ma, Chintan Patel, Edith Schonberg, Robert Schiaffino, Kavitha Srinivas

achille, aaronk, ediths, ksrinivs @us.ibm.com , malli@cn.ibm.com
chintan.patel@dbmi.columbia.edu, rschiaffino @iona.edu

IBM Research

Feb. 6, 2007

SHER

- Scalable Highly Expressive Reasoner
- Can infer *implicit* information from a relational database of *explicit* knowledge using an ontology.
- Ontology – Knowledge framework
- OWL-DL language
  - Web Ontology Language
  - Description Logic
- Enables semantic retrieval

Ontology

- Logical framework for describing
  - Concepts
  - Relationships among concepts
  - Constraints on concept definitions and relations
  - Relationships of individuals to these concepts
- TBox
  - Definition of terms (concepts)
  - Relationships among terms
- ABox
  - Assertions about individuals
  - Relationships of individuals to other individuals and terms

Example: Family Ontology

T Box

$Woman \equiv Person \cap Female$ (Conjunction)

$Man \equiv Person \cap \neg Woman$ (Negation)

$Mother \equiv Woman \cap \exists hasChild.Person$ (Existential quantification)

$Father \equiv Man \cap \exists hasChild.Person$

$Parent \equiv Mother \cup Father$ (Disjunction)

$GrandMother \equiv Mother \cap \exists hasChild.Parent$

$BigFamilyMother \equiv Mother \cap \exists \geq 3 hasChild.Person$ (Cardinality)

$MotherOfSons \equiv Mother \cap \forall hasChild.Man$ (Universal quantification)

A Box

$\{Mary : Mother, Tom : Father, \langle Mary, George \rangle : hasChild\}$

NCBI Taxonomy

- Name: NCBI taxonomy
- Primary Use: ‘backbone’ for other organism-oriented data
- Host: National Center for Biotechnology Information
- Format: Proprietary
- Size: 200K nodes
- Changes
  - Primarily additions, made on an ongoing basis, ~4% terms added each quarter
  - Re-organization occurs on a curated basis – based on consensus in literature

  Example change:

  re-organization of subsumption hierarchy in an order of organisms

	2005 q1	2004 q4	2004 q3	2004 q2	2004 q1
additions	7557	7478	6954	7827	7695

http://http://www.ncbi.nlm.nih.gov/Taxonomy/taxonomyhome.html/

Why use semantic retrieval?

In the healthcare/ life sciences domain, there is a need for:

- Infectious disease control
- Clinical alerts/ decision support
- Public Health monitoring
- Clinical trials/research
- Mining scientific data

Emergence of Standards

- Ontologies in healthcare/life sciences (e.g.,):
  - SNOMED
  - Gene Ontology
  - Biopax

  Provides:
  - Standardization of terms
  - Use of machine interpretable definitions that allow semantic retrieval of data without custom application code.

Problems with current approach

- Requires custom code, customized for each institution and each problem
- Difficult to build and maintain custom application code, as new lab tests, new drugs, get added.
- Results in expensive errors, because of misses due to coding errors/omissions.

Custom Coding Example

Monitoring staph infection , i.e., patients who have tested positive for staph requires *hardcoding for many institution specific lab tests:*

EVENT 111 – Hospital A's Lab test for staph

EVENT 222 – Hospital B's Lab test for staph

…

Standardization of terms in an ontology helps.

However, standardization is not sufficient

Screening for Staphylococcus aureus using individual SNOMED concepts will *miss records classified at different levels of granularity:*

50269000	Staphylococcus aureus ss. anaerobius
113961008	Staphylococcus aureus ss aureus
115329001	Methicillin resistant Staphylococcus aureus
404679009	Glycopeptide resistant Staphylococcus aureus
404680007	Vancomycin resistant Staphylococcus aureus
406576009	Vancomycin intermediate/resistant Staphylococcus
406605001	Glycopeptide intermediate Staphylococcus aureus
406606000	Glycopeptide intermediate/resistant Staphylococcus aureus
406962002	Vancomycin intermediate Staphylococcus aureus

How does semantic retrieval help?

Ontology definitions can be used to automatically infer correct matches to a concept without custom code. E.g. for staph infection screening, match at the top level concept:

Get all patients who had a lab test for Staphylococcus aureus

A semantic retrieval system like SHER will automatically match patient records that were coded in terms of all other concepts that are also Staph.

Reasoning Tasks

- Consistency checking
  - TBox
    - Concept definition which includes $C \sqcap \neg C$
  - ABox
    - Individual whose concept set includes $C \sqcap \neg C$
    - Individual with constraints $\leq \mathbf{m}R \sqcap \geq nR$ $(m>n)$
    - Merger of disjoint individuals
  - Central task; subsumes all others

- Implied class hierarchy
  - $C \sqsubseteq D$; $D \sqsubseteq C$; $C = D$; C unrelated to D
  - $C \sqsubseteq D \leftrightarrow C \sqcap \neg D$ is unsatisfiable

- Membership of individuals in classes
  - $a{:}C \leftrightarrow$ Adding $a{:}\ \neg C$ to the ABox creates an inconsistency

Query answering

- Queries in DL are all reduced to satisfiability checks which is not scalable for ontologies with a large number of instances.
  - Subsumption query:

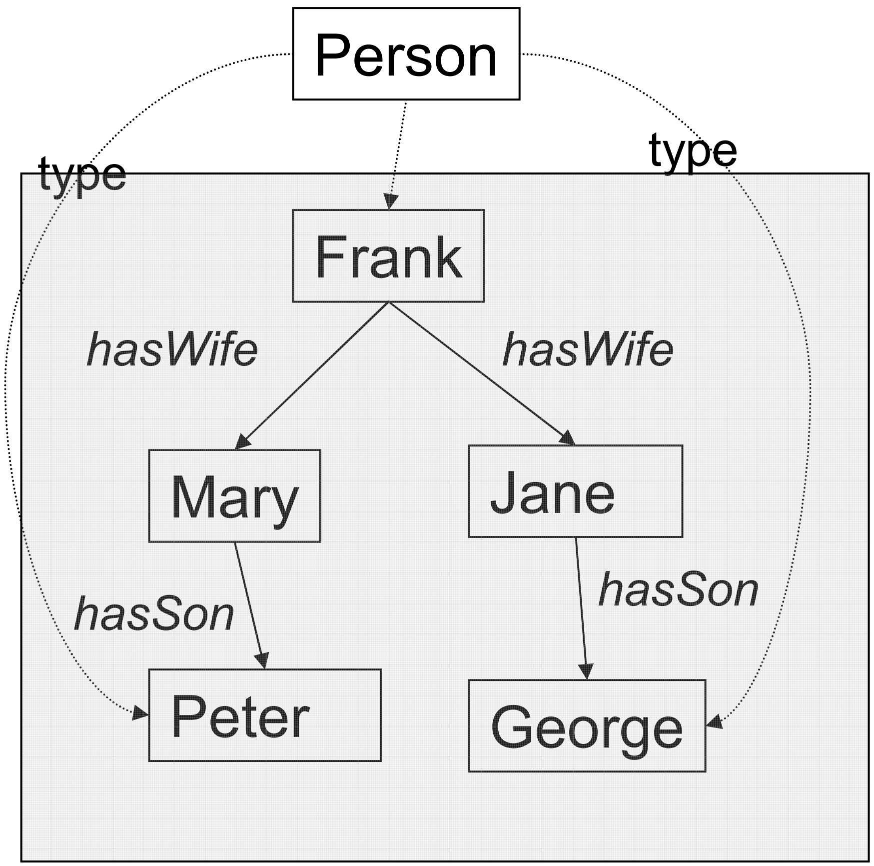
$$C \subseteq D \ iff \ C \bigcap \neg D \ is \ NOT \ satisfiable$$

  - Instance query:

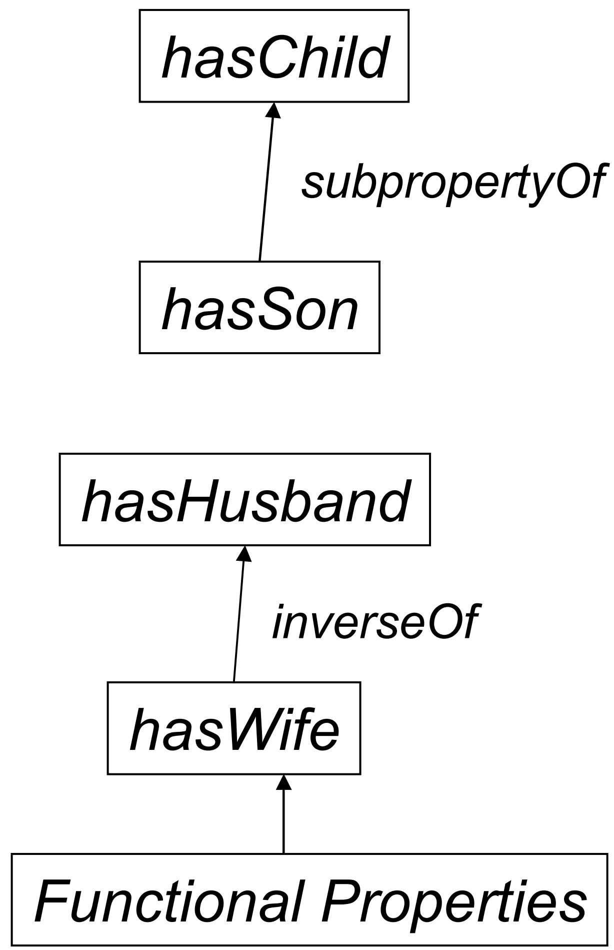
$$Tom : Person \ iff \ \ Tom : \neg Person \text{ is NOT satisfiable}$$

Inferencing for querying

Relations Query: hasChild Mary ?x



Answer: Peter, George



Ontology Language Expressiveness: OWL-DL

- Concepts (e.g., C, D)
- Roles (e.g. P,Q,R,R^-)
  - objectProperty
  - functionalProperty
  - symmetricProperty
  - transitiveProperty
  - datatypeProperty
  - inverseOf
- Restrictions
  - Existential – someValuesFrom
  - Universal – allValuesFrom
  - Cardinality – minCardinality, maxCardinality, cardinality
- Concept Hierarchy
  - equivalentClass , subclassOf
- Complex Concepts
  - intersection , union , negation
- Individuals
  - Assertions: R<a,b> where a and b are individuals and R is a role
- Nominals
  - Concepts with one or more specific individuals

OWL Ontology semantics

- Inheritance
  - Concepts inherit restrictions from subsuming concepts
  - Roles inherit restrictions from subsuming roles
- No unique name assumption
  - Individuals with different names are not assumed to be different individuals
    - Two edged sword
      - Allows for more inference
      - May lead to undesired inferences
- Open world assumption
  - We assume additional assertions can be added to the ABox
  - Thus, we do not infer anything from the absence of an assertion
    - $C = \leq nR$ and c has fewer than n type R assertions does not imply c:C
  - Holds in reality in some types of ontologies but not all

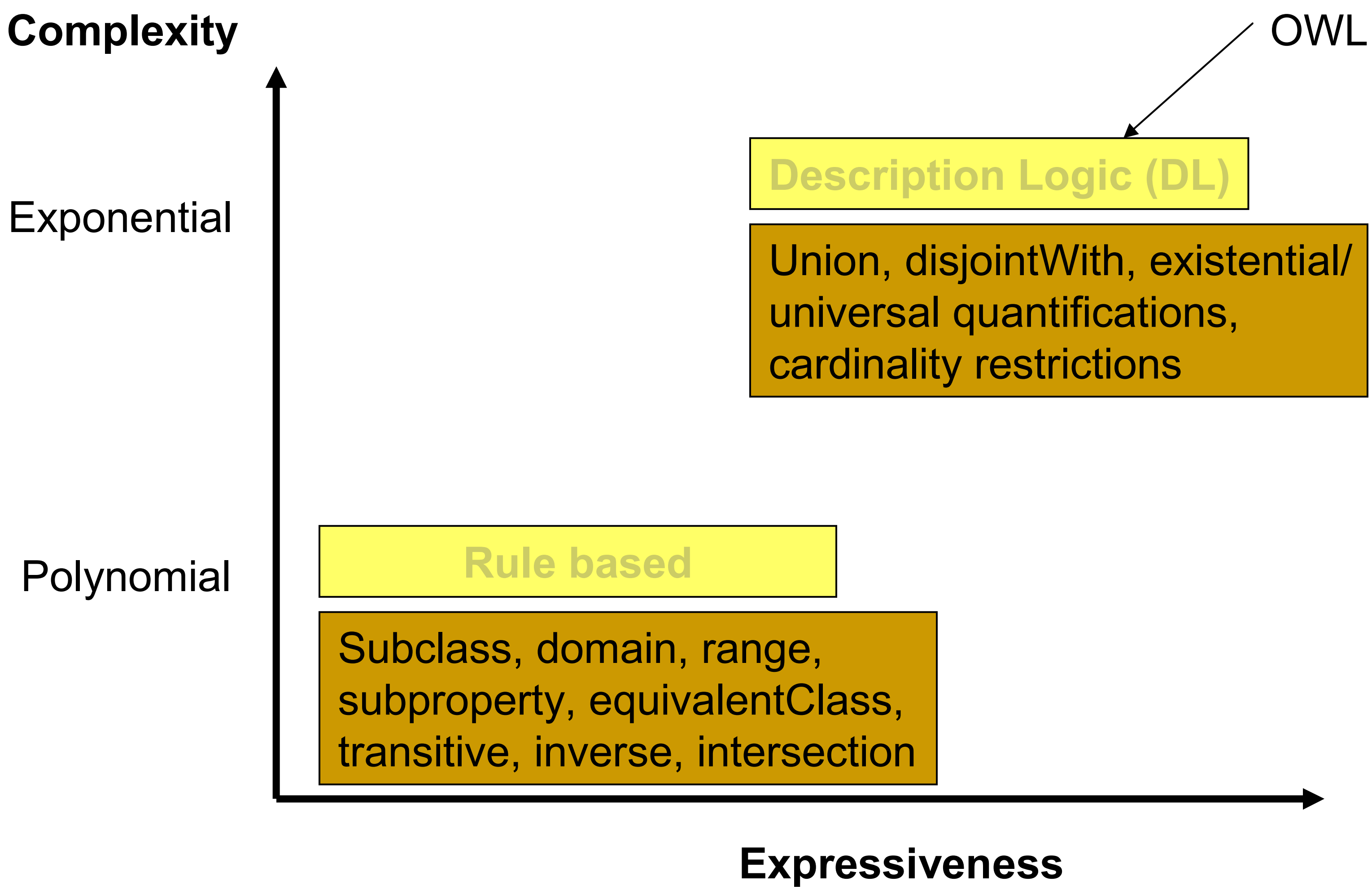
Problem – Scalable inferencing/querying of ontologies

No existing reasoners that scale up to large ontologies

- Computational complexity of reasoning
- Inconsistencies in ontologies.
- Inadequate query answering in expressive ontologies.

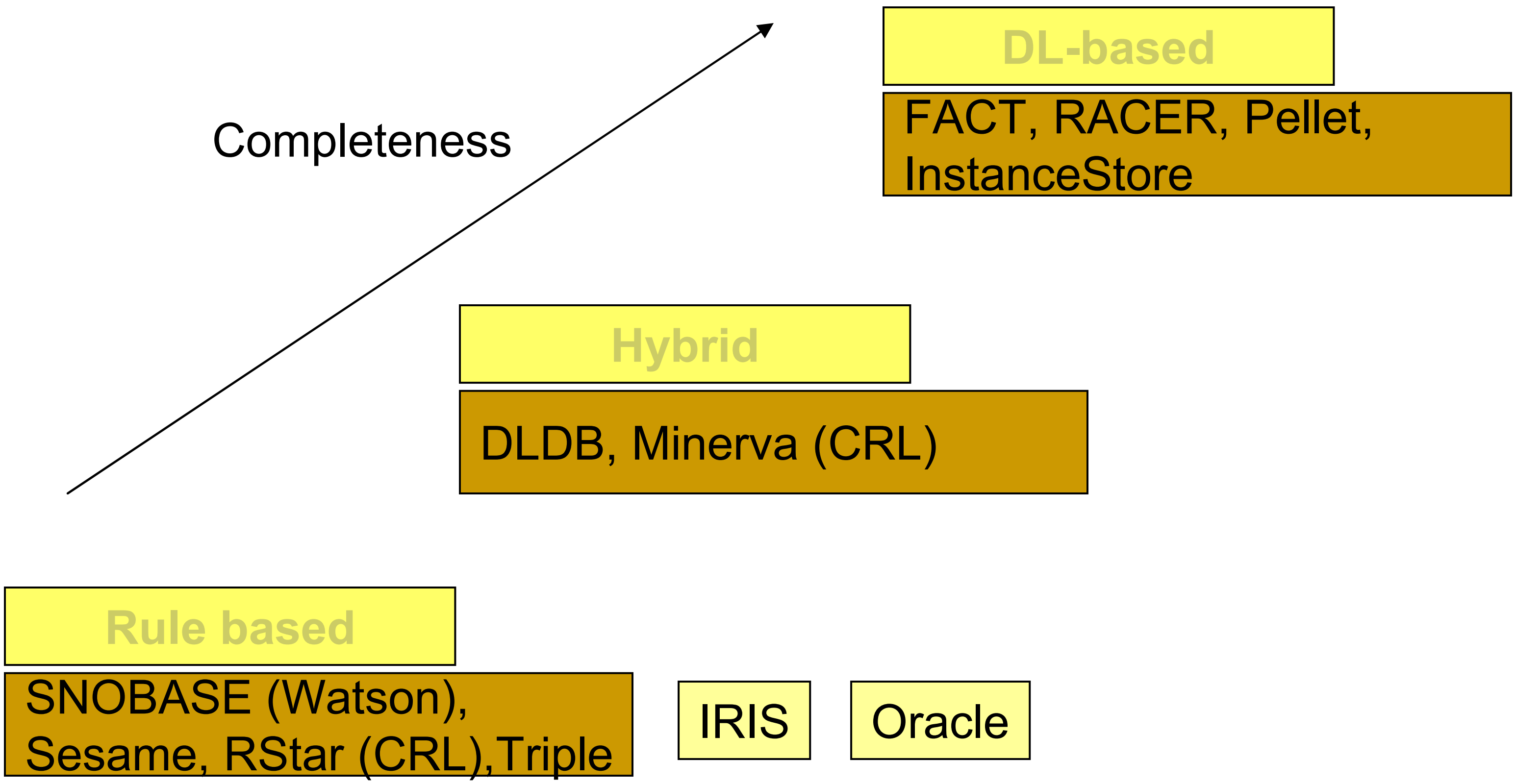
No reasoners that can deal with rapid changes in ontologies.

Computational complexity in reasoning



State of the art - Summary

Knowledge compilation: **All inferences materialized for the ontology upon load; rapid change means re-inferencing.**



Overview of our approach

- Prune parts of the ontology not relevant to the reasoning task at hand
  - Concepts ; roles ; assertions
- Summarize the ontology replacing "isomorphic" concepts and individuals by a single representative
- Partition the reduced ontology
- Persist the ontology in a DBMS
- Use DBMS queries to extract relevant parts of the ontology
- Create an in-memory image (graph) of each ontology segment if possible
- Reason over in-memory images when possible
- Reason over the DBMS representation when necessary

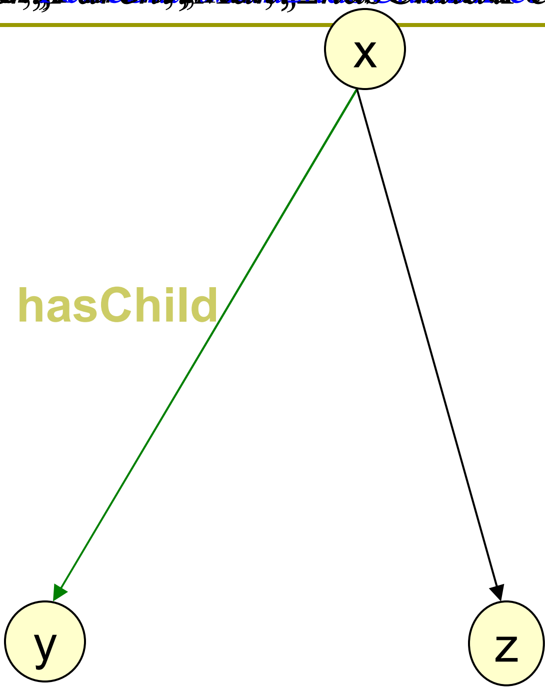
The tableau algorithm

- Verify that there is at least one consistent interpretation for the ABox and for each concept.
- Non-deterministic (due to disjunction and cardinality constraints)
- Unfold each concept for an individual in terms of the concepts defining it (completion graph)
- Can either show that each concept set of an individual C is satisfiable (one path without clash) or that ¬C is unsatisfiable

Example: Tableaux expansion

$Father \sqcap Mother$

$L(x) = \{Father, Mother, Parent, Man, \exists hasChild.Person, Person, \neg Woman, \neg(Person \sqcap \neg Female)\}$



$L(y) = \{Person\}$

Clash! ......

$L(z) = \{Father, Mother, Parent, Man, \exists hasChild.Person, Person, \neg Woman, \neg Person \sqcup \neg Female), \neg Person\}$

Tableau rules for ABox consistency

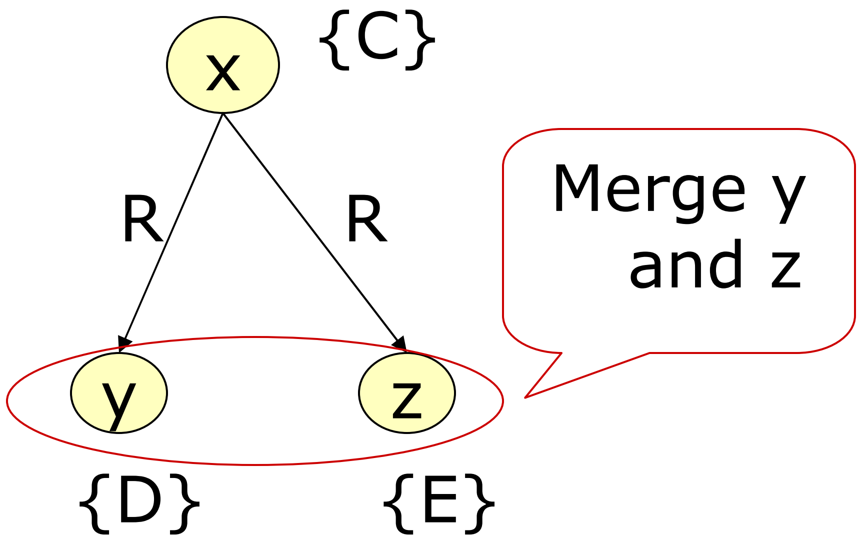
- ⊓-rule: If a:C⊓D
  - Add a:C and a:D if they are not both already present.
- ⊔ -rule: If a:C ⊔ D
  - Add a:C or a:D if neither is not already present (non-deterministic)
- ∃-rule: If a:∃R.C
  - Add x ; R<a,x> ; x:C
- ∀-rule: If a:∀R.C ; R<a,b>
  - Add b:C
- ≤-rule: If R<a,b> ; R<a,c> ; a:≤1R
  - Merge b and c
  - Generalizes to ≤nR with appropriate disjunction (non-deterministic)
- ≥ rule: If R<a,b> ; R<a,c> ; a:≥nR and fewer than n R<a,b>
  - Add n R<a,b>
- ∀ and ≤-rules are global. The others are not.

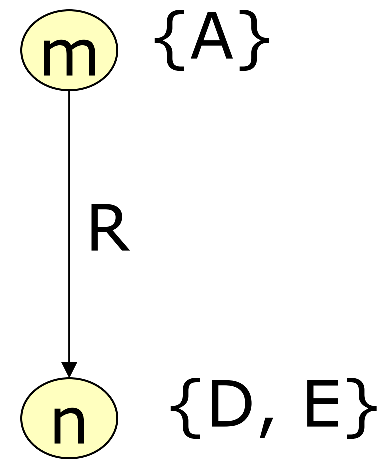
Testing consistency in large ABoxes

- Inconsistencies are due to contradictory concepts at the same node.
- Concepts can flow from one node to another in the completion graph.
- Thus, in general we have to consider the entire graph when reasoning
- Some types of roles give rise to a flow of concepts. (Global roles)
- Some do not (Local roles)
- We can break down the consistency checking task to checking individual concept satisfiability and the more general effects of global roles.
- The first task is easy. The second, in general is not.
- But, we can ignore local roles in performing the second task.

Example: Complexity in TBox reasoning

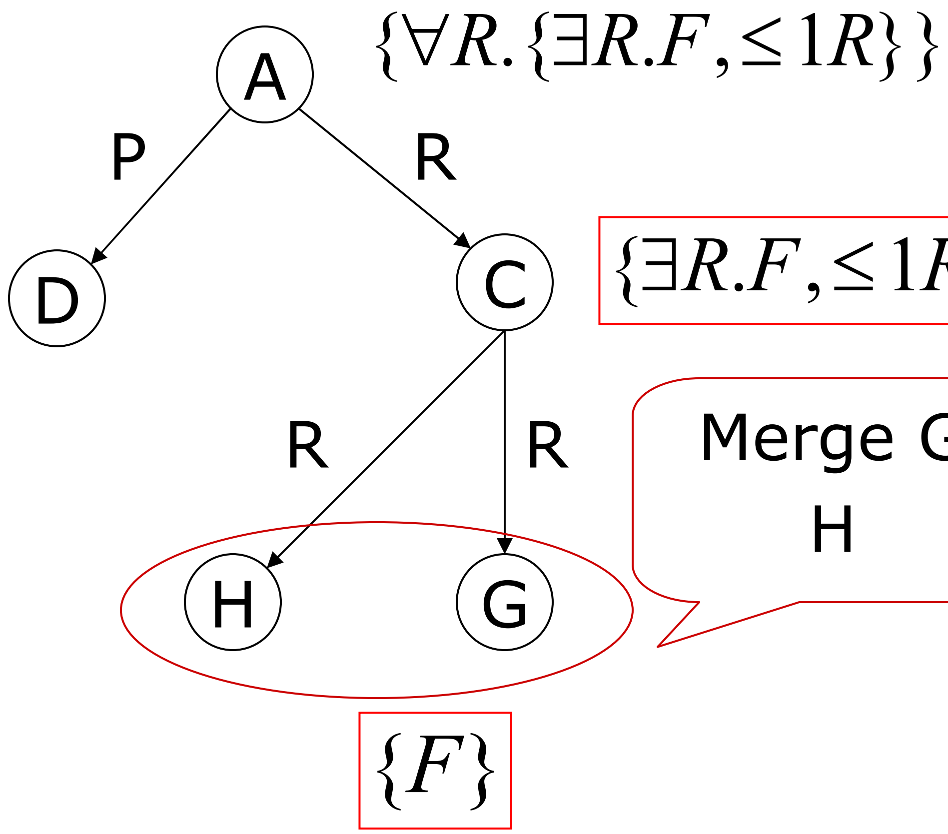
$$C \equiv \exists R.D \cap \exists R.E \cap \leq 1R$$
$$A \equiv \exists R.(D \cap E) \cap \leq 1R$$





C is equivalent to A (because of the cardinality restriction)

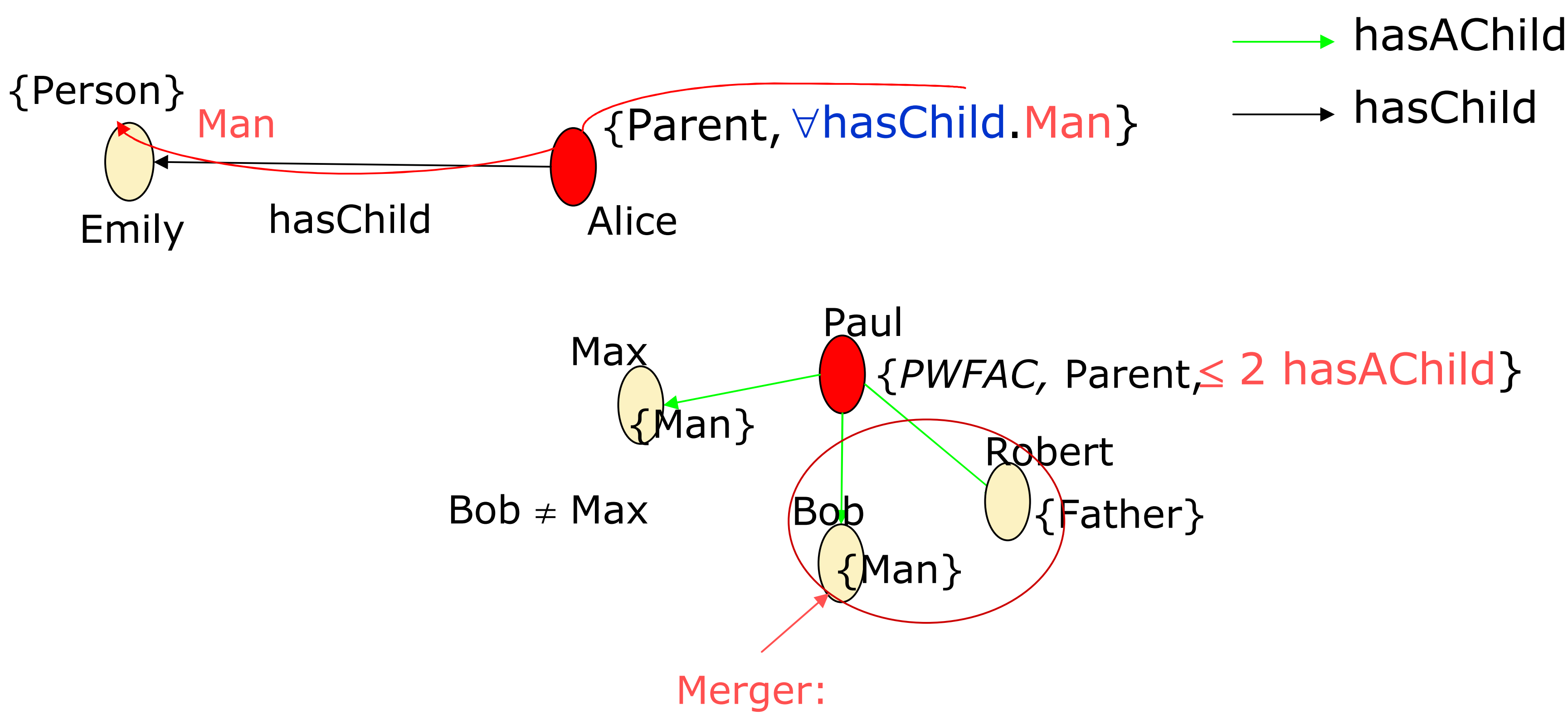
Complexity in ABox reasoning



Note that concepts migrate from A to C and then to the merged node

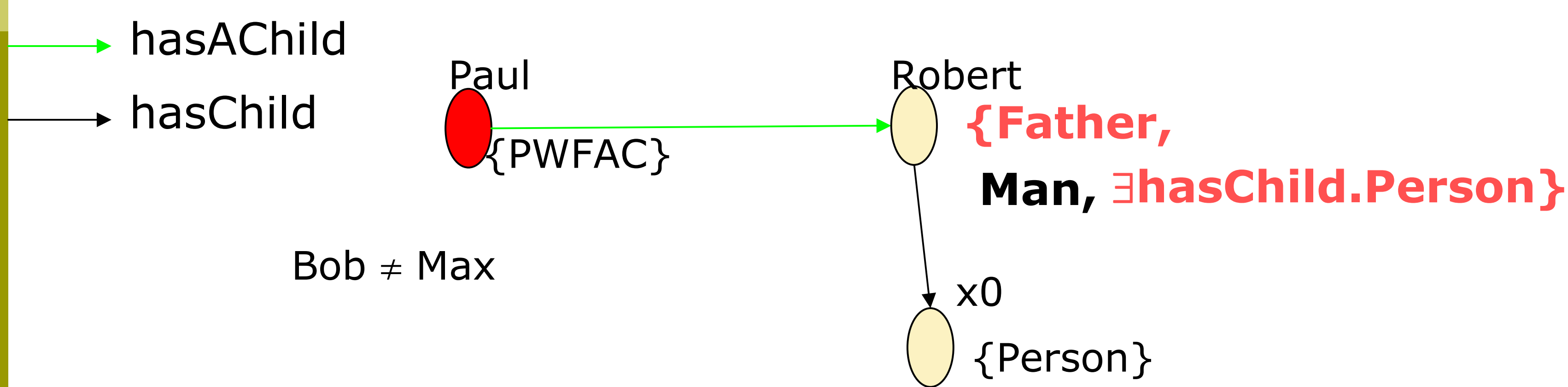
Global Effects in Reasoning

- Global Effect (GE) rules
  - affect other preexisting individuals
    - ∀ & ≤ rules
    - Propagation of new concepts through *Global Effect role assertion*



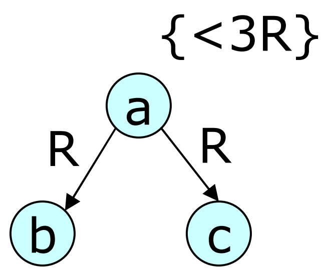
Local Effects in Reasoning

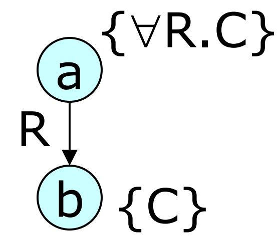
- Local Effect (LE) rules
  - No effect on other preexisting individuals
    - $\sqcap$, $\sqcup$, $\exists$ & $\geq$ rules
- Local Effect role assertion:
  - not involved in global effect rule application
  - can safely be removed
- How to determine that assertion R(a, b) is a LE role assertion?



Pruning

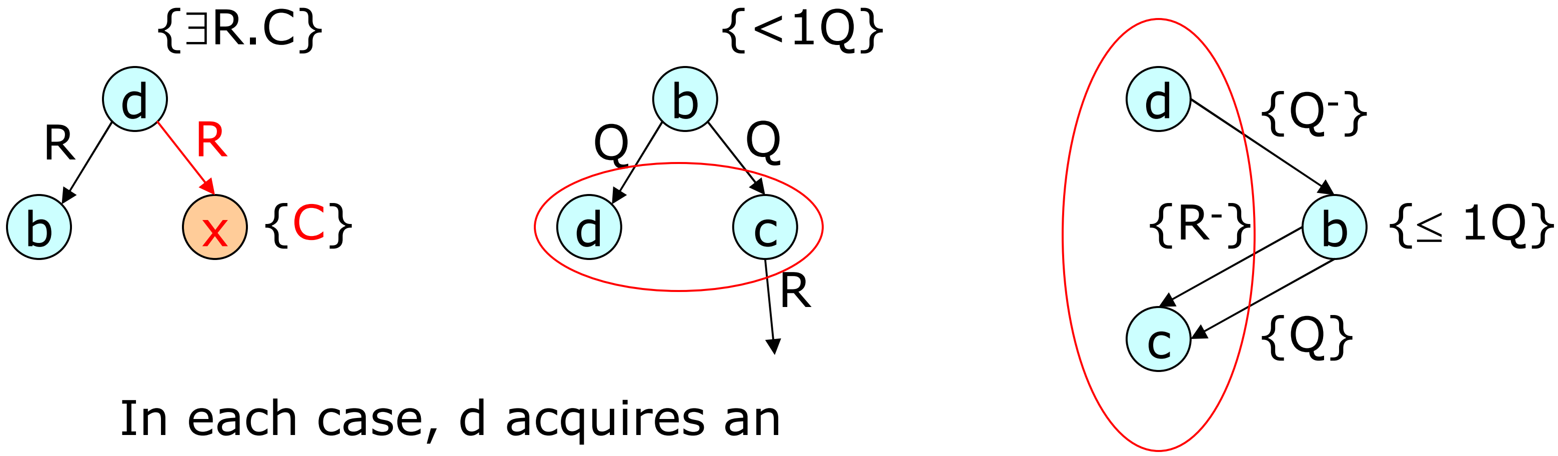
- Remove roles only involved in Local Effects
- Remove roles where the global effects they are involved in cannot trigger tableau rules
- Ignore the propagation of known concepts





Pruning

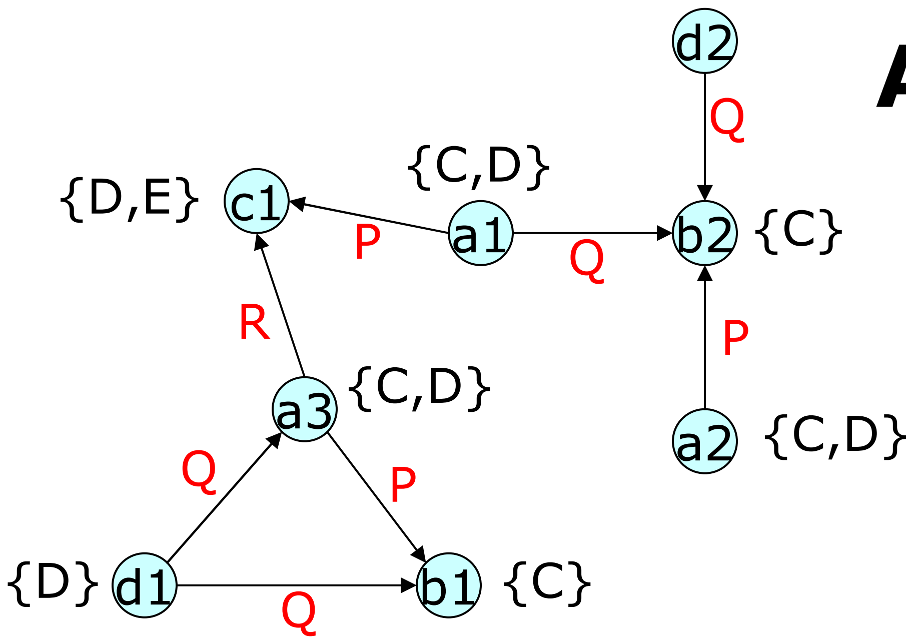
- Determining that no mergers can take place at a node c with a min-cardinality constraint $\leq$ nR requires that we be able to determine the number of R-neighbors of c.
- Determining the exact number of R-neighbors can be complex
- Our algorithm computes an upper bound



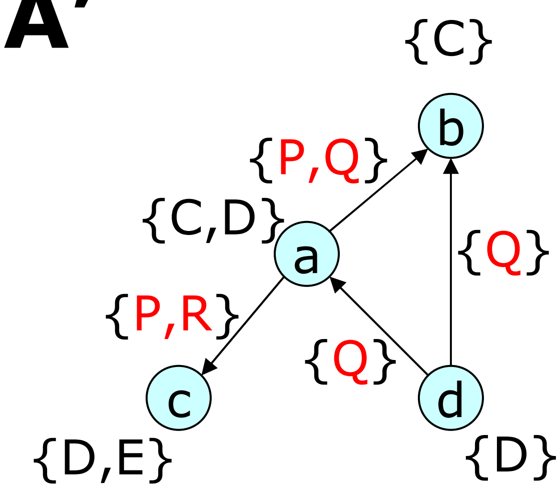
In each case, d acquires an R-neighbor

Summary Graph

- In general, many individuals have the same concept sets associated with them
- These individuals also often have the same, or at least similar, roles associated with them
- We can dramatically reduce the size of the ABox by representing each such set of individuals by a single individual



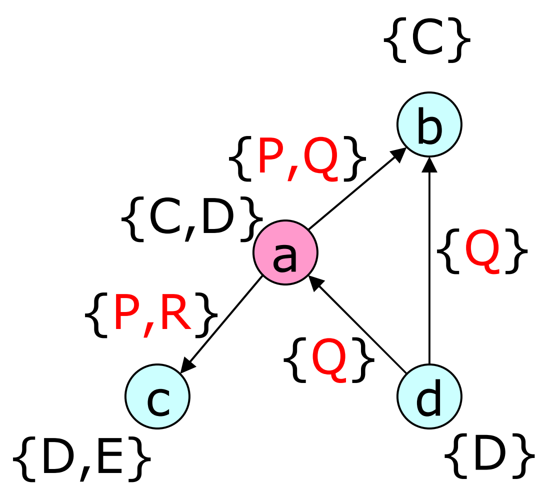
A→A'



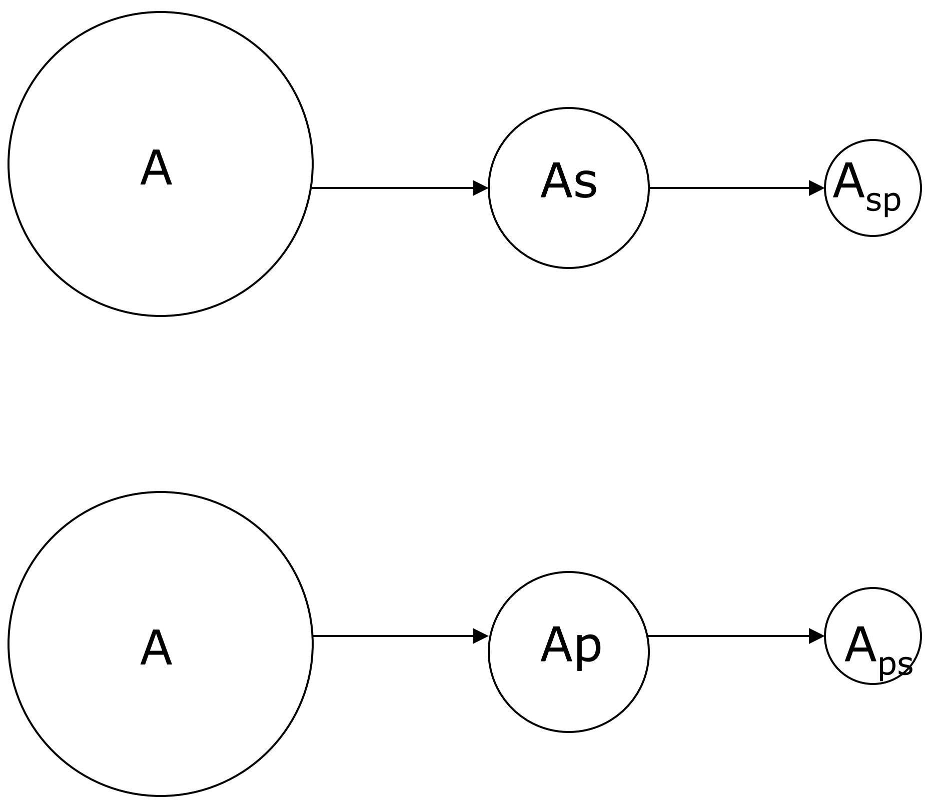
Summary Graph

- If the Summary Graph is consistent, then the original ontology must be
- But the original graph may be consistent while the Summary Graph is not because edges from multiple individuals are adjacent to nodes in the summary graph
- E.g., if P is a functionalProperty, there is a clash at node a even though there is none in the original ontology

A′



Two possible approaches



We can summarize
and then prune
- OR -
We can prune
and then summarize

Two possible approaches

- If pruning is more effective than summarization, doing pruning first would be more efficient
- If we are doing multiple queries against the ontology (i.e., defining a new concept and searching for individuals that satisfy it), doing summarization first could be more efficient if we only have to do it once.
- We have implemented both approaches.

Ontology complexity

- Generalized class inclusions
  - OWL-DL only allows atomic classes to appear explicitly on the left hand side of an expression
  - But the same atomic class may appear on the left hand side of more than one expression
- This is equivalent to allowing complex expressions to appear on the left hand side of an expression
  - $A = C \sqcap D$
  - $A \sqsubseteq B$
  - $\rightarrow C \sqcap D = B \sqcap C \sqcap D$

Ontology complexity

- Cardinality constraints
  - Maximum cardinality
    - $\leq nR$
    - Leads to non-deterministic mergers
  - Minimum cardinality
    - $\geq nR$
    - Leads to role generation
  - Cardinality
    - Equivalent to both minimum and maximum
- Disjunction
  - $A = B \sqcup C$
    - Leads to non-determinism and alternation

Ontology complexity

- Interacting functional properties
  - $A \sqsubseteq \exists P.C \sqcap \exists Q.D$
  - $(P \sqsubseteq R) \sqcap (Q \sqsubseteq R) \sqcap (\leq 1R)$
  - $\rightarrow A \sqsubseteq \exists (P,Q).(C \sqcap D)$
- Interacting universal and existential properties
  - $A \sqsubseteq \exists P.C \sqcap \forall Q.D$
  - $(P \sqsubseteq Q)$
  - $\rightarrow A \sqsubseteq \exists P.(C \sqcap D) \sqcap \forall Q.D$
- Negation
  - Leads to reasoning over infinite sets

Ontology complexity

- If many of these types of complexity are simultaneously present or if any are present in the ABox in too great a quantity, the problem becomes truly intractable.
- Fortunately, in this case it also becomes very hard to understand and so most real ontologies do not suffer so greatly from these problems as to make them unapproachable.
- Thus, it makes sense to consider algorithms that are sound and complete but which have high worst case complexity and it is not always necessary to limit ourselves to relatively inexpressive ontologies.
- Proper design of an ontology can make it tractable where a poorly designed ontology of the same size and expressiveness is not.

Designing good ontologies

- A side benefit of our investigation has been we have identified factors that make ontologies harder to analyze
- Many of these factors also make ontologies harder to understand and work with
  - Propagation of concepts through deeply nested restrictions
  - Concepts defined in terms of many restrictions
  - Interactions among restrictions (e.g., functionalProperties)
  - Large variety of patterns used to define concepts
  - Functional properties

Designing good ontologies

- Suppose we have:
  - $B \sqsubseteq A$; $C \sqsubseteq A$; $D \sqsubseteq A$
  - Disjoint(B,C) ; Disjoint(B,D) ; Disjoint(C,D)
- Consider the effect of adding
  - $A = B \sqcup C \sqcup D$
  - Or: $A = B \sqcup C \sqcup D \sqcup \text{Other}$
- We have "closed" A; i.e.
  - $B = A \sqcap \neg C \sqcap \neg D$; etc.
- This can make the ontology easier to analyze
- More importantly, it may we what we really meant.

Designing good ontologies

- Suppose we have:
  - Name type FunctionalProperty
- Do we really mean
  - We want every individual to have at most one name and it is an error if any individual has more than one.
  - Or: Individuals can have more than one name. Merge individuals inferred by other means to be the same but with different names.
- We may want to answer this question differently for different roles.

Designing good ontologies

- Do we mean
  - $A = \exists P.C$ -or- $A \sqsubseteq \exists P.C$
- The first is a definition, allowing us to infer class membership
- The second is a constraint, preventing us from inferring class membership by other means

Designing good ontologies

- Subcategorization
  - Hierarchical
    - Color = Red ⊔ Yellow ⊔ Blue
    - Red = PaleRed ⊔ NormalRed ⊔ DeepRed ; etc.
  - Orthogonal
    - Color = Red ⊔ Yellow ⊔ Blue
    - ColorDepth = Pale ⊔ Normal ⊔ Deep
- The latter is preferable when it is appropriate, but this is not always possible; e.g.:
  - BodyPart = Bone ⊔ Fluid ⊔ …
  - Bone = LongBone ⊔ ShortBone
  - Fluid = Lymph ⊔ Blood

Designing good ontologies

- The more the ontology states explicitly, the clearer the meaning and the less likely an unintended inference will take place
- The less the ontology states explicitly, the more succinct and structured it is.

Retrieval of Legal Information

- Relevant statutes , cases , decisions
- Available both in hard copy and on-line
- Currently a major industry
  - West , Lexis-Nexis
  - On-line search and retrieval
  - Publishing
  - Multibillion dollar annual revenue
- Major consumer of time and costs for all those who practice law

Analysis of Legal Information

- Aids for preparing legal briefs
  - Abstracting cases and decisions
- E-discovery
  - Machine-readable information obtained during the pre-trial discovery process
- Tools for current suppliers of legal information
  - West , Lexis-Nexis
  - Case/decision categorization/annotation
- A potential next phase for this effort

Current State of the Art

- Case Law Print Sources
  - Digest system
    - Published Cases , case blurbs
    - Organized by court and regionally
    - Subject-specific digests (e.g., education law)
  - Secondary sources
    - Restatements
    - American Law Reporter
    - American Law Institute publications
  - Legal periodicals
    - Law reviews
    - Journals
- Similar aids for retrieving statutes
- Also available on-line

Current State of the Art

- Manual retrieval is time consuming
- On-line retrieval with search engines is expensive
  - West and Lexis-Nexis often charge $600-$1000 / hour
  - While faster than manual search, still time consuming and labor intensive
- West and Lexis-Nexis use an army of lawyers to read, index, categorize and abstract legal information which is constantly changing
  - A ruling or statute which supersedes a prior one invalidates the prior one
  - Legal information differs by jurisdiction (federal, individual states) multiplying the effort
- Concept-based retrieval is done only on a limited basis
  - There is no significant evidence that reasoning is done

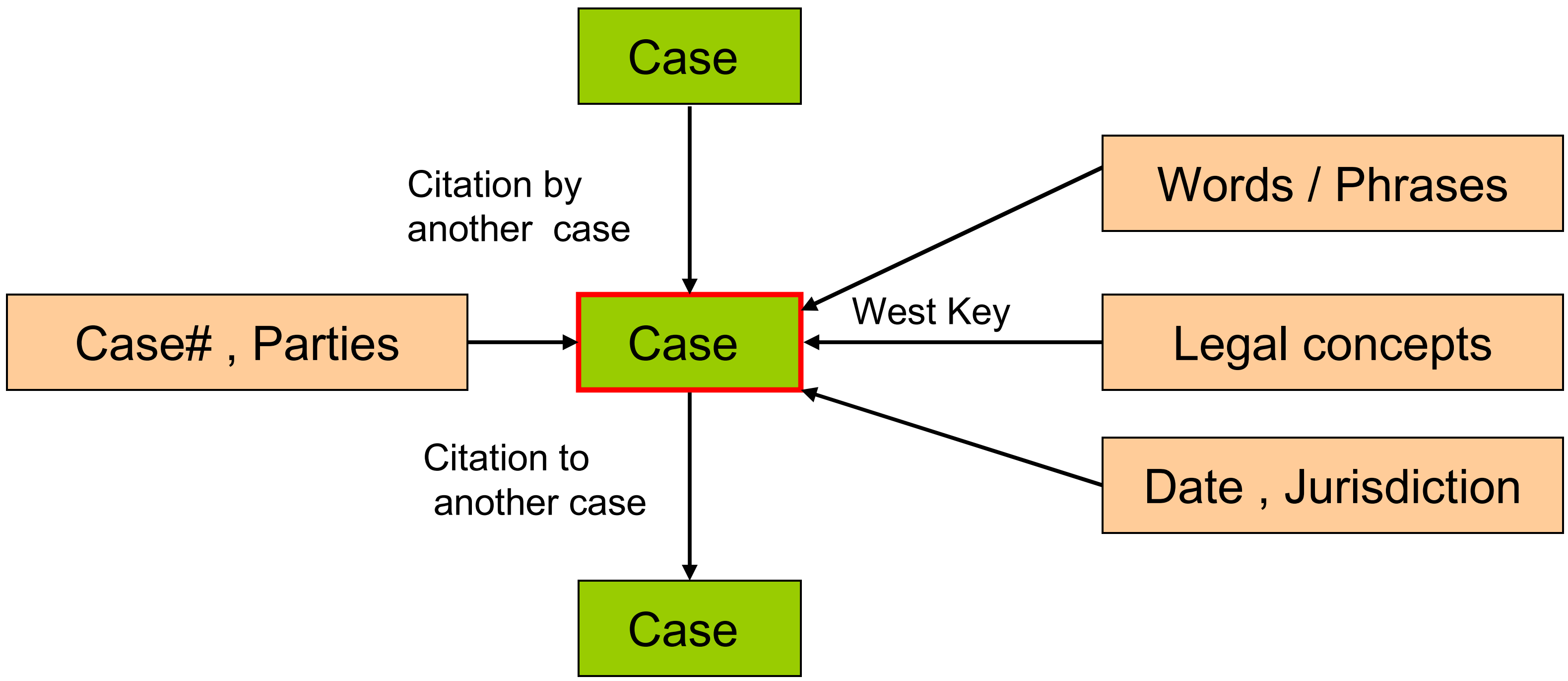
The Retrieval Process

- ACE – legal search tools must be
  - Accurate , Complete and Efficient
- Near total recall is essential
- Low precision increases time and effort
- Manual
  - The lawyer , clerk or librarian goes to a library and looks for relevant statutes , cases and decisions using the existing research aids
  - Potentially relevant information must be read, copied, analyzed and abstracted
    - The farther this process goes, the greater the cost and effort
    - The sooner a document is rejected, the more likely relevant information may be lost
    - Less thorough search
      - Could lead to rejection of claims
      - Could even be considered malpractice

On-Line Retrieval

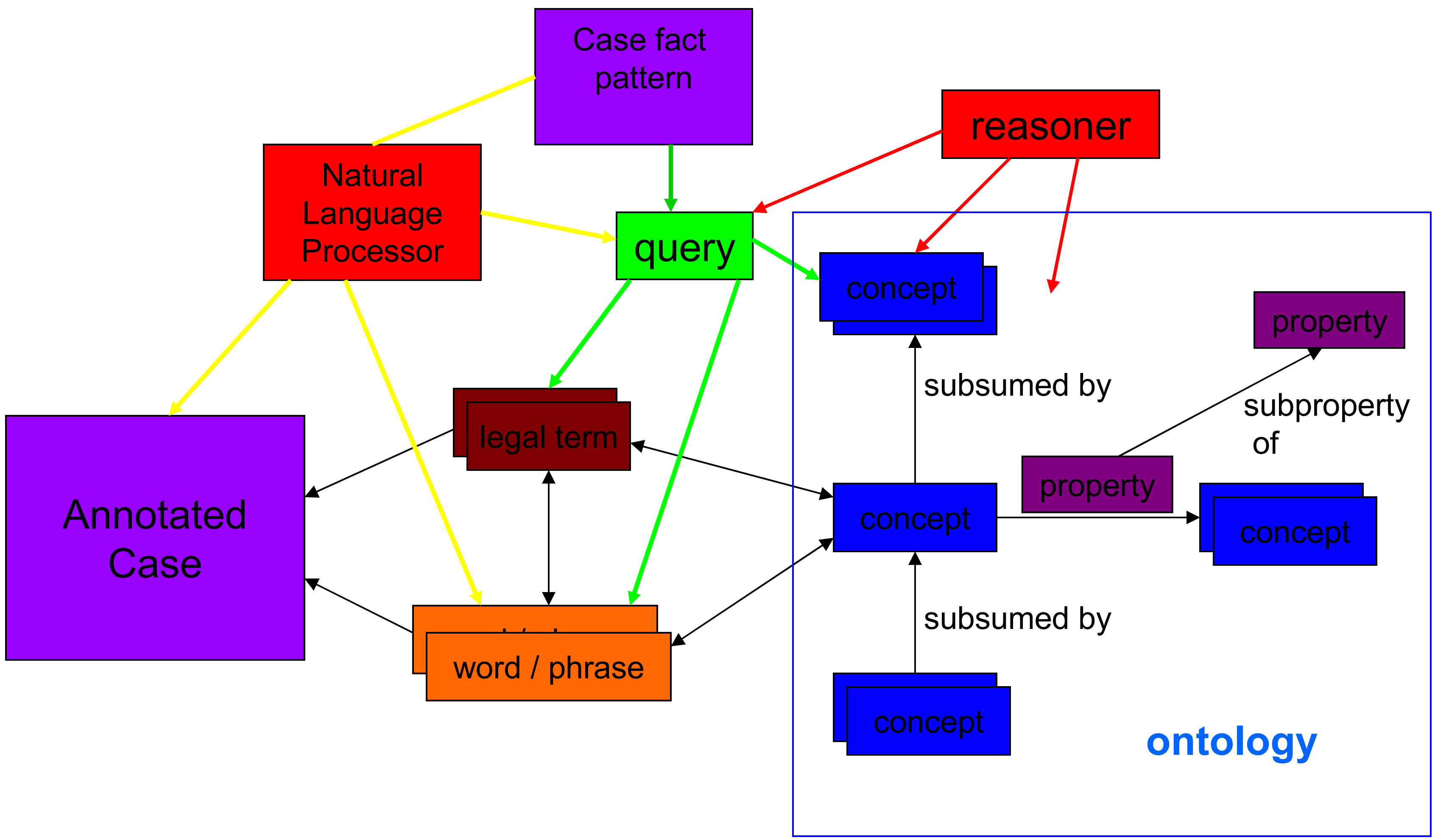
- Cases organized by
  - Jurisdiction
  - Chronologically
  - Case number
  - Parties
  - Roughly 80,000 legal concepts
- Keyword search
  - Words and phrases
  - Extended by suggested terms
  - Extended by morphology

Citations



"see" Smith vs. Jones – Support for a claim
"but see" – Contrasting opinion
Parentheticals – Reason for citation
Not totally reliable

Conceptual (Semantic) Search



Fact Pattern for a Case

- A truck explodes on a highway, injuring a nearby driver
  - The truck was transporting sodium.
  - The driver was operating the truck in a reasonable manner.
- Actionable?
  - Does the driver have a claim against the trucking company?

Typical Results of "Traditional" On-Line Search

- Search for "torts", "injury", etc. produce thousands of cases within the jurisdiction
- Searches for "transporting" or "dangerous materials" produces thousands of cases
- Search for "sodium" -AND- "explosion" –AND- "truck" produces 4 cases, some relevant, some not.
- Lawyer patiently searches for 4 hours and finds some relevant cases, some relating to trains, some relating to other volatile materials.

Reasoning

Paraphrased excerpt from Chapter 21 of a torts restatement:

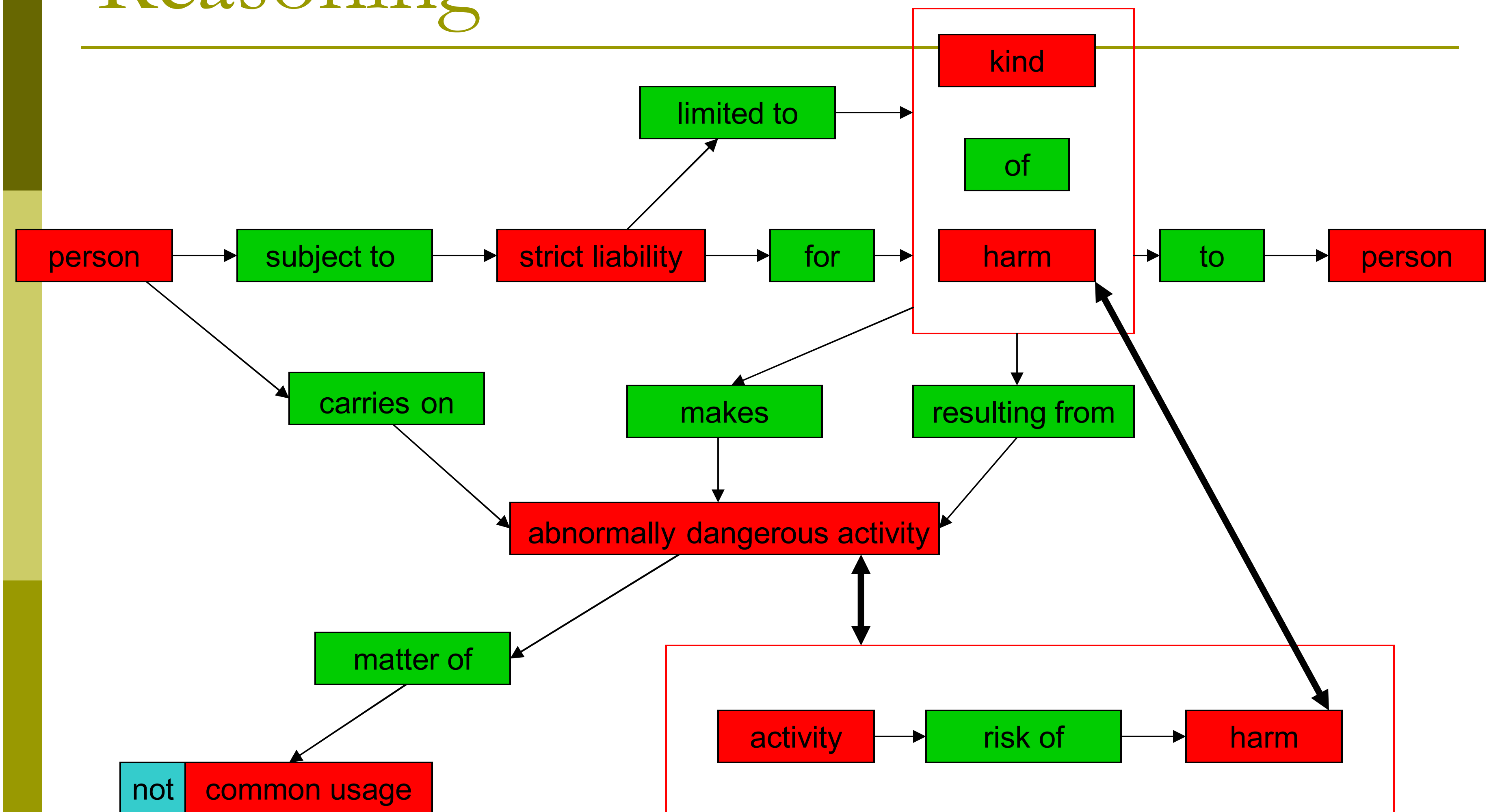
One who carries on an abnormally dangerous activity is subject to strict liability for harm to the person … resulting from the activity, although he has exercised … care to prevent the harm. This strict liability is limited to the kind of harm … which makes the activity abnormally dangerous.

Abnormally Dangerous Activities

In determining whether an activity is abnormally dangerous, the following factors are to be considered:

- (a) existence of a high degree of risk of … harm to the person …
- (b) extent to which the activity is not a matter of common usage

Reasoning

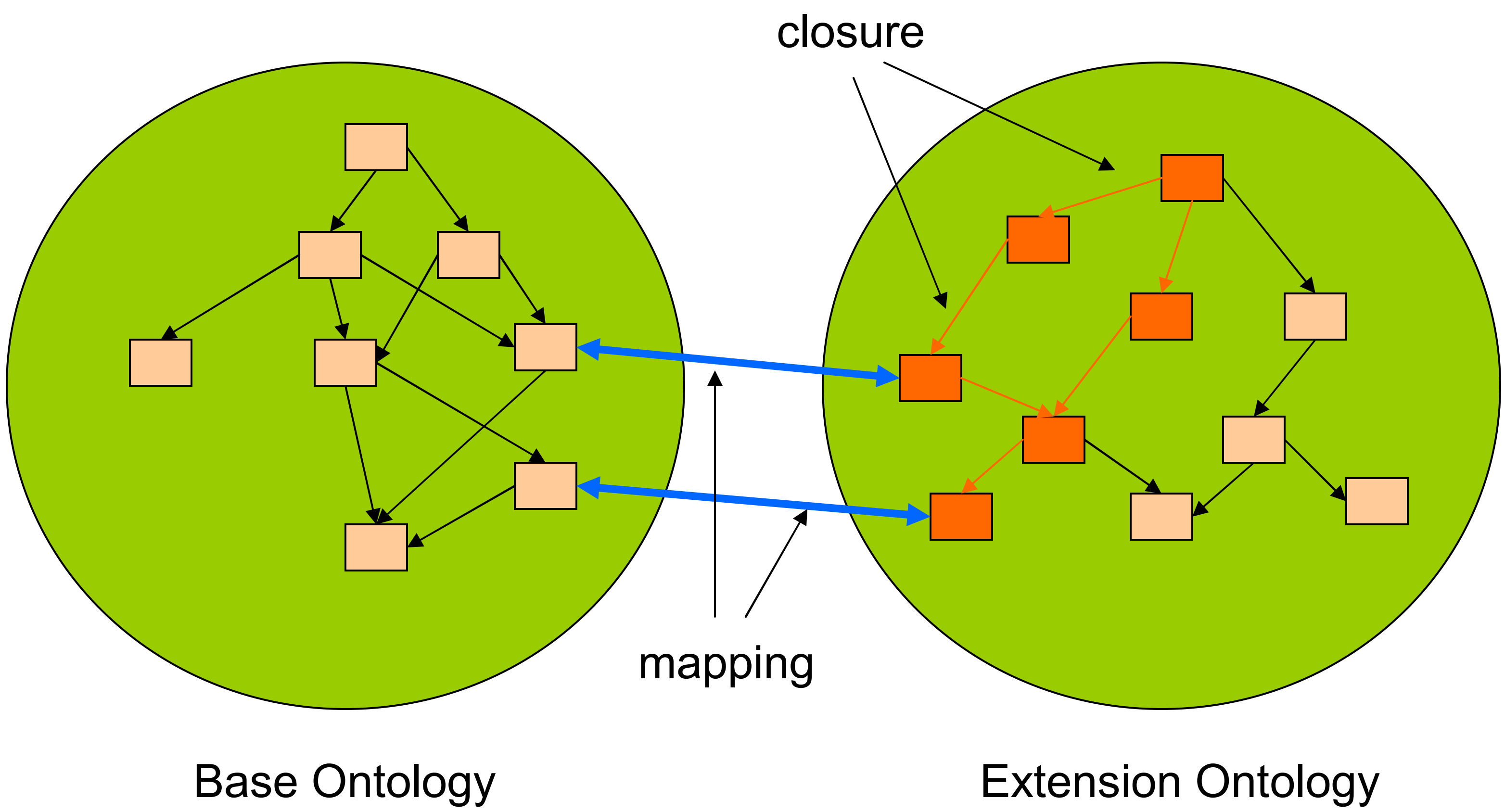


kind
of
harm
limited to
person
subject to
strict liability
for
to
person
carries on
makes
resulting from
abnormally dangerous activity
matter of
not
common usage
activity
risk of
harm

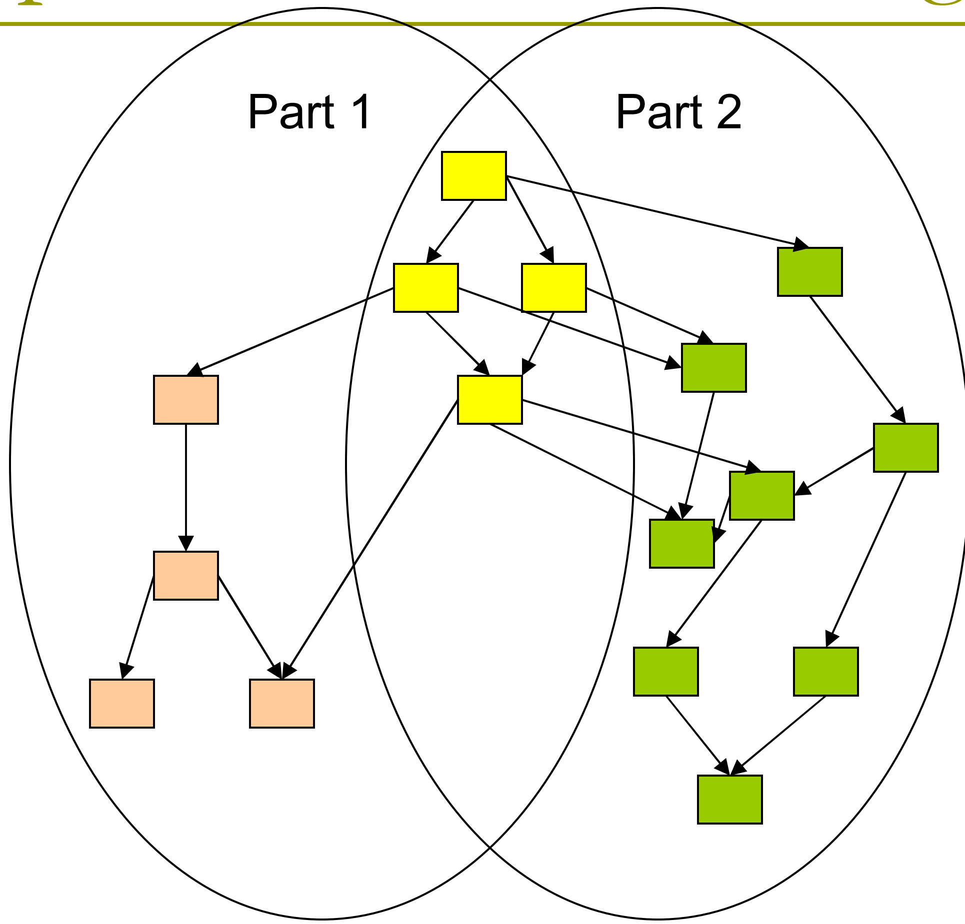
Concept-Based Retrieval

- The truck was engaged in an abnormally dangerous activity based on the fact that sodium is dangerous.
- Strict liability includes situations that do not involve negligence
- The harm came from the explosion of the sodium
- The trucking company is liable
- Any case (in that jurisdiction) which relates to harm caused by transporting hazardous materials could be relevant to this case

Closure of an Ontology with Respect to Another Ontology



Decomposition of an Ontology



Summarization effectiveness

Ontology	Instances	Role Assertions	I	R A
Biopax	261,149	582,655	81	583
LUBM-1	42,585	214,177	410	16,233
LUBM-5	179,871	927,854	598	35,375
LUBM-10	351,422	1,816,153	673	49,176
LUBM-30	1,106,858	6,494,950	765	79,845
NIMD	1,278,540	1,999,787	19	55
ST	874,319	3,595,132	21	183

I – Instances after summarization
RA – Role assertions after summarization

Conclusions

- New heuristics for scaling reasoning over large ABoxes in secondary storage
  - Static analysis of OWL ontologies
  - Summarization technique
- Dramatic reduction in time and space requirements for 4 realistic very large Aboxes.
- Future Work
  - More accurate static analyses
  - Extension to datatypes and nominals
  - Concept flow analysis in TBox and ABox