

Database Relations

Parent		Ancestor	
p	c	x	y
'bob'	'ned'	'bob'	'ned'
'jim'	'bob'	'jim'	'bob'

0. Construct Database Relations (project 3 code)

Schemes:
Parent(p,c)
Ancestor(x,y)

Facts:
Parent('bob','ned').
Parent('jim','bob').
Ancestor('bob','ned')
Ancestor('jim','bob')

Solved Body Predicates

Ancestor		Parent	
A	P	P	C
'bob'	'ned'	'bob'	'ned'
'jim'	'bob'	'jim'	'bob'

1. Solve Body Predicates

Note: body predicates can have variables or constants (just like queries)! For this case, we only have variables

Rules:
Ancestor(A,C):-Ancestor(A,P),Parent(P,C).

Queries:
Ancestor(x,'ned')?

2. Join Body Predicates

Joined Relation

No Name		
A	P	C
'jim'	'bob'	'ned'

3. Project Columns Specified by Head Predicate

Projected, Joined Relation

No Name	
A	C
'jim'	'ned'

4. Rename with Database for Union Compatibility

Renamed, Projected, Joined Relation

Ancestor	
x	y
'jim'	'ned'

4a. Grab Name from Head Predicate

4b. Rename to Header from DB Relation

5. Union with DB Relation

Updated Database Relations

Parent		Ancestor	
p	c	x	y
'bob'	'ned'	'bob'	'ned'
'jim'	'bob'	'jim'	'bob'
		'jim'	'ned'

6. A tuple was added, so we re-evaluate the rules again until we don't add any more tuples (fixed point algorithm)

Note: For this case, no new tuples will be added the second time around

7. After fixed point algorithm finishes (no more tuples added to DB on ANY of the rules), evaluate queries as normal (project 3 code)

Evaluated Queries

Ancestor	
x	y
'bob'	
'jim'	