Homework #3 Answers,
Information Integration, CSCI 6967-01
Due February 21, 2008 at 2pm

Question 1. Let's use the skolem approach. For simplicity, I will rename the attributes in the views below.

(1)$r(f_1(W1), W1) : -v_1(W1)$
(2)$p(X2, Y2, f_2(X2, Y2)) : -v_2(X2, Y2)$
(3)$p(X31, f_3(X31, W31), f_4(X31, W31)) : -v_3(X31, W31)$
(4)$s(X32, f_5(X32, W32), W32)) : -v_3(X32, W32)$

Given the following query:

$Q1 : ans(X, K, L) : -p(X, Y, K), r(Y, Z), s(K, L, Z)$

We use the only available rules for r and s to get:

$Q1 : ans(X, K, L) : -p(X, Y, K), r(Y, Z), s(K, L, Z)$
$\qquad Y = f_1(Z), W1 = Z$
$Q1 : ans(X, K, L) : -p(X, f_1(Z), K), v_1(W1), s(K, L, Z)$
$\qquad X32 = K, W32 = Z, L = f_5(K, Z)$
$Q1 : ans(X, K, f_5(K, Z)) : -p(X, f_1(Z), K), v_1(W1), v_3(K, Z)$

At this point, we know that there is no way to answer this query as one of the requested attributes in the head is now a skolem function. If you wanted to continue beyond this point anyways, here is what would happen. This rule would not join with rule 3 since f_1 and f_3 do not match. We join with rule 2 instead to get:

$Q1 : ans(X, K, f_5(K, Z)) : -p(X, f_1(Z), K), v_1(W1), v_3(K, Z)$
$\qquad X2 = X, Y2 = f_1(Z), K = f_2(X, f_1(Z))$
$Q1 : ans(X, f_2(X, f_1(Z)), f_5(K, Z)) : -v_2(X, f_1(Z)), v_1(W1), v_3(f_2(X, f_1(Z)), Z)$

This query is not executable since two views now have skolems which we cannot check.

Question 2. Do the same as Question 1 for the following view definitions and queries:

$(1)p(X11, f_1(X11, W11), f_1(X11, W11)) :- v_1(X11, W11), context(W11 > 5)$
$(2)r(f_1(X12, W12), W12) :- v_1(X12, W12), context(W12 > 5)$
$(3)r(X21, Y21) :- v_2(X21, Z21, Y21, W21), context(Y21 < 10, Z21 \neq W21)$
$(4)s(X22, Z22, W22) :- v_2(X22, Z22, Y22, W22), context(Y22 < 10, Z22 \neq W22)$
$(5)p(X31, Y31, f_2(X31, Y31)) :- v_3(X31, Y31), context(f_2(X31, Y31) > 5)$
$(6)p(X41, f_3(X41, W41), f_4(X41, W41)) :- v4(X41, W41)$
$(7)s(42, f_4(X42, W42), W42) :- v4(X42, W42)$

Given $Q1$:

$Q1 : ans(X, K, L) :- p(X, Y, Z), r(Y, Z), s(K, L, Z), Z > 10$
$(1)X11 = X, Y = Z = f_1(X, W11)$
$Q1 : ans(X, K, L) :- v_1(X, W11), r(f_1(X, W11), f_1(X, W11)), s(K, L, f_1(X, W11)), f_1(X, W11) > 10,$
$context(W11 > 5)$
(2) no unifier, fail, (3) $X21 = Y21 = f_1(X, W11)$
$Q1 : ans(X, K, L) :- v_1(X, W11), v(f_1(X, W11), Z21, f_1(X, W11), W21),$
$s(K, L, f_1(X, W11)), f_1(X, W11) > 10, context(W11 > 5, Y21 < 10, Z21 \neq W21)$
(2) skolem in view, fail, restart

$Q1 : ans(X, K, L) :- p(X, Y, Z), r(Y, Z), s(K, L, Z), Z > 10$
$(5)X31 = X, Y31 = Y, Z = f_2(X, Y)$
$Q1 : ans(X, K, L) :- v_3(X, Y), r(Y, f_2(X, Y)), s(K, L, f_2(X, Y)), f_2(X, Y) > 10, context(f_2(X, Y) > 5)$
(2,3) both will lead to a skolem in view, fail, restart

$Q1 : ans(X, K, L) :- p(X, Y, Z), r(Y, Z), s(K, L, Z), Z > 10$
$(6)X41 = X, Y = f_3(X, W41), Z = f_4(X, W41)$
$Q1 : ans(X, K, L) :- v_4(X, W31), r(f_3(X, W41), f_4(X, W41)), s(K, L, f_4(X, W41)), f_4(X, W41) > 10$
(2) will fail to unify and (3) will lead to a skolem in view, fail, no rewritings

Given $Q2$:

$Q2 : ans(X, W) :- p(X, Y, Z), r(Z, W)$
$(1)X11 = X, Y = Z = f(X, W11)$
$Q2 : ans(X, W) :- v_1(X, W11), r(f(X, W11), W), context(W11 > 5)$
$(2)W12 = W, W11 = W$
$Q2 : ans(X, W) :- v_1(X, W), context(W > 5)$

This is a rewriting of this view. Unfolding we get:

$Q2' : ans(X, W) :- p(X, Y, Y), r(Y, W), W > 5.$

We can map $Q2$ to this query by setting $Z = Y$, hence this query $Q2'$ is contained in query $Q2$.

$Q2 : ans(X, W) :- p(X, Y, Z), r(Z, W)$
$(5)X31 = X, Y31 = Y, Z = f_2(X, Y)$
$Q2 : ans(X, W) :- v_3(X, Y), r(f_2(X, Y), W), context(f_2(X, Y) > 5)$

(2) fails to unify, $(3) X21 = f_2(X,Y), Y21 = W$

$Q2 : ans(X,W) :- v_3(X,Y), v_2(f_2(X,Y), Z21, W, W21), context(...)$

skolem in view, fail.

$Q2 : ans(X,W) :- p(X,Y,Z), r(Z,W)$

(6) leads to a similar problem as above, (2) will not unify and (3) leads to a skolem in view. Hence, no additional rewritings.

Question 3. A bit contrived? Suppose you have two types of users, humans and robots. You know that robots answer to any illogical question is 'no' and to any logical question is yes, whereas humans answer all questions either yes or no regardless of whether they are illogical or not. However, we know that humans never answer the same illogical questions as robots.

So, we have that

$type(E, robot), answer(Q, no, Qtype, E), answer(Q', A', Qtype, E') \rightarrow type(E', robot)$

Given the following views:

$v_1(E) :- type(E, robot), E < 50$
$v_2(Q, A, E) :- answer(Q, A, Qtype, E)$

where v_1 contains ids of some robots and v_2 only contains questions and answers (but not whether the question is illogical (Qtype) or the answerer is human or robot (E)). The aim is to find all the robots.

$Q_1 : ans(E) :- type(E, robots)$

Now, we can find illogical questions answered by a robot using v_1. Now, if anybody else shares an illogical question with a robot, they must also be a robot. This will give us a new set of robots. We continue this until we find more questions and more robots.

Question 4. We rewrite the views using the foreign key constraints as shown below:

$(1) p'(X,Y,Z) :- p(X,Y,Z)$
$(2) p'(X, f_1(X), f_2(X)) :- r'(X,Y,Z)$
$(3) p'(X, f_1(X), f_2(X)) :- t'(X,Y)$
$(4) r'(X,Y,Z) :- r(X,Y,Z)$
$(5) t'(X,Y) :- t(X,Y)$
$(6) s'(X,Y) :- s(X,Y)$
$(7) t'(f_3(Y), Y) :- s'(Y,X)$

Disregarding the predicates with a single rule:

$Q1 : ans(X,W) :- r(X,Y,Z), s(Y,W)$

only single rewriting for each predicate. I will simply look for alternative rewritings for each predicate beyond the obvious (1) for p and (4) for t.

$Q2 : ans(X) :- p'(X,Y,Z), t'(X,Y), s(Y,Z)$

$(2) Y = f_1(X), Z = f_2(X)$

$Q2 : ans(X) :- r(X, Y1, Z1), t'(X, f_2(X)), s(f_2(X), Z)$

Not executable anymore, will lead to a skolem in view.

similar result for using (3).

$Q2 : ans(X) :- p'(X, Y, Z), t'(X, Y), s(Y, Z)$

$(7) X = f_3(Y)$

$Q2 : ans(f_3(Y)) :- p'(f_3(Y), Y, Z), s'(Y, X1), s(Y, Z)$

Not executable anymore, will lead to a skolem in view.

As a result, the only feasible rewriting is the one obtained by using p, t, s. As the join predicates are the same for $Q3$, there is also a single rewriting for this query.