CSCI 548 – Homework 5

1. Write the Local-As-View (LAV) rules for each source respectively.

S1(title, name) -> Artwork (title, name, creation_date, dimension, type), Artist (name, nationality, birth_date, death_date, biography), death_date < 1930, nationality = “American”

S2(title, creation_date) -> Artwork (title, name, creation_date, dimension, type), Artist (name, nationality, birth_date, death_date, biography), creation_date < 1950, nationality = “American”, name = “Walt Disney”, type = “cartoon”

S3(title, name) -> Artwork (title, name, creation_date, dimension, type), creation_date > 1990, type = “painting”

S4(title, URL) -> Image (title, URL, tag), Artwork (title, name, creation_date, dimension, type), type = “cartoon”

2. Given the query that searches for all the cartoons created by American artists before 1940 that have image URL online. The returned results should contain the titles of the cartoons, artist names and image URL. Write the query using the mediated scheme and reformulate the query in LAV using the Bucket algorithm. Show the critical derivations.

Q(title, name, URL) :- Image (title, URL, tag), Artist (name, nationality, birth_date, death_date, biography), Artwork (title, name, creation_date, dimension, type), creation_date < 1940, type = “cartoon”

Using the following:
Image (title, URL, tag) -> Image (T, U, TA)
Artist (name, nationality, birth_date, death_date, biography) -> Artist (N, NA, BD, DD, BI)
Artwork (title, name, creation_date, dimension, type) -> Artwork (T, N, CD, DI, TY)

Using Bucket Algorithm, query reformulation:

Q(T,N,U) :- Image (T,U,TA), Artist (N, NA, BD, DD, BI), Artwork (T, NA, CD, DI, TY), CD < 1940, NA = “American”, TY = “cartoon”

Bucket Creation:

Image(T, U, TA)	Artist(N, NA, BD, DD, BI)	Artwork(T, N, CD, DI, TY)
S4(T, U)	S1(T’, N)	S1(T, N)
	S2(T’, CD’)	S2(T, CD)
	S4(T, U’)	

Check containments:
Q (T, N, U) :- Artist (N, NA, BD, DD, B), Artwork (T, N, CD, D, TY), Image (T, U, TA), CD<1940, TY = “cartoon”, NA =“American”

i. Combination 1:

Q1’ (T, N, U) :- S1 (T’, N), S1 (T, N), S4 (T U), CD<1940, TY= “cartoon”, NA = “American” -> S1 (T, N), S4 (T, U), CD<1940, TY = “cartoon”, NA = “American

Check Q1’ (T, N, U) ⊆ Q (T, N, U) by performing unfolding:

{ Artwork (title, name', creation_date', dimension', type') ^ Image (title, URL, tag') ^ type’ = 'cartoon' } ^ { Artist (name, nationality', birth_date', death_date', biography') ^ Artwork (title, name, creation_date', dimension', type') ^ nationality’='American' , death_date’ < 1930 } ^ { creation_date < 1940 , type = 'cartoon' , nationality ='American' } -> { Artwork (title, name, creation_date', dimension', type') ^ Artist (name, nationality', birth_date', death_date', biography') ^ nationality='American' , death_date < 1930 ^ Image (title, URL, tag') ^ type’ = 'cartoon' ^ creation_date' <1940 } ⊆ Artwork (title, name, creation_date, dimension, type), Image (title, URL, tag), creation_date < 1940 , Artist (name, nationality, birth_date, death_date, biography), type = 'cartoon' , nationality='American'.

That is, Q1’ (T, N, U) ⊆ Q (T, N, U)

ii. Combination 2:

Q2'(T, N, U) :- S1 (T', N) , S2 (T, CD) , S4 (T , U) , CD < 1940 , TY = 'cartoon' , NA='American'

Check Q2’ (T, N, U) ⊆ Q (T, N, U) by performing unfolding:

{ Artwork (title, name', creation_date, dimension', type) ^ Artist (name', nationality', birth_date', death_date', biography') ^ creation_date < 1950 ^ name = 'Walt Disney' ^ type = 'cartoon' ^nationality='American'} ^ { Artist (name, nationality', birth_date', death_date', biography') ^ Artwork (title, name, creation_date', dimension',type') ^ nationality='American' , death_date < 1930 } ^ { Artwork (title, name', creation_date', dimension', type') ^ Image (title, URL, tag') ^ type = 'cartoon' } ^ { creation_date < 1940 , type = 'cartoon' , nationality ='American' } -> { Artwork (title, name, creation_date, dimension', type') ^ Artist (name, nationality', birth_date', death_date', biography') ^ Image (title, URL, tag') ^ nationality'='American' ^ death_date' < 1930 ^ name = 'Walt Disney' ^ type = 'cartoon' ^ creation_date < 1950 } ⊆ Artwork (title, name, creation_date, dimension, type), Image (title, URL, tag), Artist (name, nationality, birth_date, death_date, biography), creation_date < 1940 , type = 'cartoon' , nationality='American'.

That is, Q2' (T, N, U) ⊆ Q(T, N, U)

iii. Combination 3:

Q3' (T, N, U) :- S1 (T', N), S4 (T, U'), S4 (T , U), CD< 1940 , TY = 'cartoon' , NA ='American' -> S1 (T', N) , S4 (T , U) , CD < 1940 , TY = 'cartoon' , NA ='American'.

Check Q3’ (T, N, U) ⊆ Q (T, N, U) by performing unfolding:

{ Artwork (title, name', creation_date', dimension', type') ^ Image (title, URL, tag') ^ type’ = 'cartoon' } ^ { Artist (name, nationality', birth_date', death_date', biography') ^ Artwork (title', name, creation_date', dimension', type') ^ nationality'='American' , death_date' < 1930 } ^ { creation_date'< 1940 , type = 'cartoon' , nationality'='American' } -> { Artist (name, nationality', birth_date', death_date', biography') ^ Artwork (title, name, creation_date', dimension', type') ^ nationality'='American' , death_date' < 1930 ^ Image (title, URL, tag') ^ type' = 'cartoon' ^ creation_date' <1940 } ⊆ Artwork (title, name, creation_date, dimension, type), Artist (name, nationality, birth_date, death_date, biography), Image (title, URL, tag), creation_date < 1940 , type = 'cartoon' , nationality='American'.

That is, Q3'(T, N, U) ⊆ Q(T, N, U).

iv. Combination 4:

Q4' (T, N, U) :- S2 (T', CD'), S1 (T , N) , S4 (T , U) , CD< 1940 , TY = 'cartoon' ,NA='American'

Check Q4' (T, N, U) ⊆ Q(T, N, U) by performing unfolding:

{ Artist (name, nationality', birth_date', death_date', biography') ^ Artwork (title, name, creation_date, dimension', type') ^ nationality'='American' , death_date' < 1930 } ^ { Artist (name', nationality', birth_date', death_date', biography') ^ Artwork (title', name', creation_date', dimension', type') ^ creation_date' < 1950 ^ name' = 'Walt Disney' ^ type' = 'cartoon' ^ nationality'='American'} ^ { Artwork (title, name', creation_date', dimension', type') ^ Image (title, URL, tag') ^ type' = 'cartoon' } ^ { creation_date < 1940 , type = 'cartoon' , nationality='American' } -> { Artwork (title, name, creation_date, dimension', type') ^ Artist (name, nationality', birth_date', death_date', biography') ^ Image (title, URL, tag') ^ creation_date < 1950 ^ name = 'Walt Disney' ^ type' = 'cartoon' ^nationality='American' ^ death_date < 1930} ⊆ Artist (name, nationality, birth_date, death_date, biography), Artwork

(title, name, creation_date, dimension, type), Image (title, URL, tag), creation_date < 1940 , type = 'cartoon' , nationality='American'.

That is, Q'(T, N, U) ⊆ Q(T, N, U).

v. Combination 5:

Q5' (T, N, U) :- S2 (T', CD'), S2 (T , CD), S4 (T , U) , CD < 1940, TY ='cartoon', NA='American' -> S2 (T, CD) , S4 (T, U), CD < 1940, TY = 'cartoon' , NA='American'.

Check Q5' (T, N, U) ⊆Q(T, N, U) by performing unfolding:

{ Artist (name', nationality', birth_date', death_date', biography') ^Artwork (title, name', creation_date, dimension', type') ^ creation_date < 1950 ^ name' = 'Walt Disney' ^ type' = 'cartoon' ^nationality'='American'} ^ { Artist (name', nationality', birth_date', death_date', biography') ^Artwork (title', name', creation_date', dimension', type') ^ creation_date' < 1950 ^ name' = 'Walt Disney' ^ type' = 'cartoon' ^nationality'='American' } ^ { Artwork (title, name', creation_date', dimension', type') ^ Image (title, URL, tag') ^ type' = 'cartoon' } ^ { creation_date < 1940 , type = 'cartoon' , nationality ='American' } ^ {Artist (name', nationality', birth_date', death_date', biography') ^Artwork (title, name', creation_date, dimension', type') ^ Image (title, URL, tag') ^ type' = 'cartoon' ^ creation_date < 1950 ^ name' = 'Walt Disney' } -> { Artwork (title, name', creation_date, dimension', type') ^ Artist (name', nationality', birth_date', death_date', biography') ^ Artwork (title, name', creation_date', dimension', type') ^ Image (title, URL, tag') ^ type' = 'cartoon' ^ creation_date < 1950 ^ name' = 'Walt Disney' } ⊆ Artist (name, nationality, birth_date, death_date, biography), Artwork (title, name, creation_date, dimension, type), Image (title, URL, tag), creation_date < 1940 , type = 'cartoon' , nationality='American'.

That is, Q5' (T, N, U) ⊆Q(T, N, U).

vi. Combination 6:

Q6'(T, N ,U) :- S2 (T', CD'), S4 (T , U'), S4 (T , U) , CD < 1940 , TY = 'cartoon' , NA='American' -> S2 (T', CD'), S4 (T , U) , CD < 1940 , TY = 'cartoon', NA ='American'.

Check Q6' (T, N, U) ⊆Q(T, N, U) by performing unfolding:

{ Artwork (title, name', creation_date', dimension', type') ^ Image (title, URL', tag') ^ type' = 'cartoon' } ^ { Artist (name', nationality', birth_date', death_date', biography') ^Artwork (title', name', creation_date', dimension', type') ^ creation_date' < 1950 ^ name' = 'Walt Disney' ^ type' = 'cartoon' ^ nationality'='American'} ^ {Artwork (title, name', creation_date', dimension', type') ^ Image (title, URL, tag') ^ type' = 'cartoon' } ^ { creation_date < 1940 , type = 'cartoon' , Nation='American' } -> { Artist (name', nationality', birth_date', death_date', biography') ^Artwork (title, name', creation_date', dimension', type') ^ Image (title, URL, tag') ^ creation_date' < 1950 ^ name' = 'Walt Disney' ^ type' = 'cartoon' ^nationality'='American'} ⊆ Artwork (title, name, creation_date, dimension, type), Artist (name, nationality, birth_date, death_date, biography), Image (title, URL, tag), creation_date < 1940 , type = 'cartoon' , nationality='American'.

That is, Q6' (T, N, U) ⊆Q(T, N, U).

From all the above combinations

Q1' (T, N, U) U Q2' (T, N, U) U Q3' (T, N, U) U Q4' (T, N, U) U Q5' (T, N, U) U Q6' (T, N, U)

⊆ Q (T, N, U).

Sources that give most containment are: S1, S2, S4

3. Write the Global-As-View (GAV) rules for each of the relations in mediated scheme respectively.

Creating a new source S5(title, type)

Artist (name, “American“,_,_,_) <- S1 (title, name), S5 (title, type)
Artist (name, “American“,_,_,_) <- S3 (title, name)
Artwork (title, name,_,_, ‘cartoon’) <- S1 (title, name), S5 (title, type)
Artwork (title,_, creation_date,_,_) <- S2 (title, creation_date)
Artwork (title,_,_,_,_) <- S4 (title, url)
Artwork (title,_,_,_,_) <- S3 (title, name)
Image (title, url,_) <- S4 (title, url)

4. Given the same query in question 2, reformulate it in GAV rules. Show the critical derivations.

Q(title, name, url) :- Artist (name, “American”, birth_date, death_date, biography) ^ Artwork (title, name, creation_date, dimension, type) ^ Image (title, URL, tag), creation_date<1940

Checking for containment. Getting:

1.

Q1’ (T, N, U) :- S1 (T, N), S5 (T, TY), S1 (T, N), S5 (T, TY), S4 (T, U), creation_date<1940, type = ”cartoon” -> S1 (T, N), S5 (T, TY), S4 (T, U), creation_date<1940, type = ”cartoon”

2.

Q2’(T, N, U) :- S1 (T, N), S5 (T, TY), S2 (T, CD), S4 (T, U), creation_date<1940, type = ”cartoon” -> S1 (T, N), S5 (T, TY), S2 (T, CD), S4 (T, U), creation_date<1940, type = ”cartoon”

Getting,

Q1’ (T, N, U) ⊆ Q2'(T, N, U), that is, Q2’ contains Q1’. Thus, we remove Q1’.

3.

Q3' (T, N, U) :- S1(T, N), S5(T,TY), S4(T, U), creation_date<1940, type=”cartoon” -> S1(T, N), S5(T, TY), S4(T, U), creation_date<1940,type='cartoon'

Getting,

Q3’ ⊆Q2’, that is, Q2’ contains Q3’. Thus, we remove Q3’.

4.

Q4' (T ,N, U) :- S1(T, N), S5(T, TY), S3(T, N), S4(T, U), creation_date<1940

For Q4’, S3 (T, N) contains the artwork created after 1990. Thus we remove Q4’.

5.

Q5'(T, N, U) :- S3(T, N), S1(T, N), S5(T, TY), S4(T, U), creation_date<1940, type='cartoon'

S3 (T, N) contains the artwork created after 1990. Thus we remove Q5’.

6.

Q6'(T, N, U) :- S3(T, N), S2(T, CD), S4(T, U), creation_date <1940

S3 (T, N) contains the artwork created after 1990. Thus we remove Q6’.

7.

Q7' (T, N, U) :- S3(T, N), S4(T, U), creation_date <1940