

Report July 19th

1. Umberto's comments

During this week Umberto answered some questions concerning the algorithm. We started by **“Should the algorithm create one PCD for each concrete service or one PCD for each abstract service?”**

He explained that we should have one PCD for each abstract service to maximize the number of possible combinations. However, if we have a dependency between abstract services a PCD for more than one abstract service is created, and also the coverage domain of the PCD includes both abstract services. Considering this I showed him the example I tested before:

Query:

Q(disease, patients, dnas, birth, gen, addr, wheatherstatistic) :- A1(disease, patients),
A2(patients,dnas), A3(patients, birth, gen, addr), A4(addr, wheatherstatistic)

Concrete services:

S2(disease, patientsssn, dna) :- A1(disease, patientsssn), A2(patientsssn, dna)
S3(patientsssn, dna) :- A2(patientsssn, dna)
S4(patientsssn, dateofbirth, gender, address) :- A3(patientsssn, dateofbirth, gender, address)
S5(address, wheatherstaticalinformation) :- A4(address, wheatherstaticalinformation)
S6(patientsssn, diseases) :- A5(patientsssn, diseases)
S7(patientsssn, vaccinateddiseases) :- A6(patientsssn, vaccinateddiseases)

The **rewriting results** were:

1) Q(disease,patients,dnas,birth,gen,addr,wheatherstatistic) :-
S2(disease,patients,-),S2(-,patients,dnas),S4(patients,birth,gen,addr),S5(addr,wheatherstatistic)
2) Q(disease,patients,dnas,birth,gen,addr,wheatherstatistic) :-
S2(disease,patients,-),S3(patients,dnas),S4(patients,birth,gen,addr),S5(addr,wheatherstatistic)

In the result (1) the service S2 is called two times, and in (2) S1 and S3 are called together. **“Why do we have these answers if we have the dependency between A1 and A2 in S2?”**

The rewriting answers are correct. Note that we have all variables in the head of the service definition, because of that we do not have dependency between A1 and A2, and two different PCDs are created for this service.

2. New Example

Considering Umberto's comments I tried to produce a new example expressing the dependencies between abstract services.

Query:

Q(disease, patients, dnas, birth, gen, addr, wheatherstatistic) :- A1(disease, patients),
A2(patients,dnas), A3(patients, birth, gen, addr), A4(addr, wheatherstatistic)

Concrete services:

S1(disease, patientsssn) :- A1(disease, patientsssn)
S2(disease, dna) :- A1(disease, patientsssn), A2(patientsssn, dna)
S3(patientsssn, dna) :- A2(patientsssn, dna)

S4(patientsssn, dateofbirth, gender, address) :- A3(patientsssn, dateofbirth, gender, address)
 S5(address, wheatherstaticalinformation) :- A4(address, wheatherstaticalinformation)
 S6(patientsssn, diseases) :- A5(patientsssn, diseases)
 S7(patientsssn, vaccinateddiseases) :- A6(patientsssn, vaccinateddiseases)

Note that I removed *patientsssn* from the head of the service S2 and I supposed that now this variable creates a dependency between the abstract services A1 and A2. However the rewriting answer was:

Q(disease,patients,dnas,birth,gen,addr,wheatherstatistic) :-
 S1(disease,patients),S3(patients,dnas),S4(patients,birth,gen,addr),S5(addr,wheatherstatistic)

And the PCDs created were:

S1 [A1(disease,patients)]
 S3 [A2(patients,dnas)]
 S4 [A3(patients,birth,gen,addr)]
 S5 [A4(addr,wheatherstatistic)]

No answer using S2 was created and also no PCD for it. I also tried to remove S1, but I got an error running the algorithm.

3. Example with SLA

Query:

Q(disease, patients, dnas, birth, gen, addr, wheatherstatistic) :- A1(disease, patients),
 A2(patients,dnas), A3(patients, birth, gen, addr), A4(addr, wheatherstatistic){ cost \leq 1\$,
 location = close, response time < 2s, data provenance = any}

I added in the end of the query, between braces, the user's requirements that should be after matched in the SLAs. I put them between braces to differentiate from the constraints that could also be part of the query. I am supposing that I have access to the services SLAs. Considering the concrete services below, the PCDs created are:

Concrete services:

S1(disease, patientsssn) :- A1(disease, patientsssn)
 S2(disease, patientsssn) :- A1(disease, patientsssn)
 S3(patientsssn, dna) :- A2(patientsssn, dna)
 S4(patientsssn, dateofbirth, gender, address) :- A3(patientsssn, dateofbirth, gender, address)
 S5(address, wheatherstaticalinformation) :- A4(address, wheatherstaticalinformation)
 S6(patientsssn, diseases) :- A5(patientsssn, diseases)
 S7(patientsssn, vaccinateddiseases) :- A6(patientsssn, vaccinateddiseases)

PCD 1:

S S1

SLA { cost = 0.2, location = close, response time < 2s, data provenance = fresh }

h { disease \longrightarrow disease, patientsssn \longrightarrow patientsssn }

φ { disease \longrightarrow disease, patientsssn \longrightarrow patients }

G A1(disease, patientsssn)

Def \emptyset

has_opt false

PCD 2:

S S2

SLA { cost = 0.4, location = close, response time < 3s, data provenance = fresh }

h { disease \longrightarrow disease, patientsssn \longrightarrow patientsssn }

φ { disease \longrightarrow disease, patientsssn \longrightarrow patients }

G A1(disease, patientsssn)

Def \emptyset

has_opt false

PCD 3:

S S3

SLA { cost = 0.1, location = close, response time < 1s, data provenance = fresh }

h { patientsssn \longrightarrow patientsssn, dna \longrightarrow dna }

φ { patientsssn \longrightarrow patients, dna \longrightarrow dnas }

G A2(patientsssn, dna)

Def \emptyset

has_opt false

PCD 4:

S S4

SLA { cost = 0.1, location = close, response time < 1s, data provenance = fresh }

h { patientsssn \longrightarrow patientsssn, dateofbirth \longrightarrow dateofbirth, gender \longrightarrow gender, address \longrightarrow address }

φ { patientsssn \longrightarrow patients, dateofbirth \longrightarrow birth, gender \longrightarrow gen, address \longrightarrow addr }

G A3(patientsssn, dateofbirth, gender, address)

Def \emptyset

has_opt false

PCD 5:

S S5

SLA { cost = 0.1, location = close, response time < 1s, data provenance = fresh }

h { address \longrightarrow address, wheatherstaticalinformation \longrightarrow wheatherstaticalinformation }

φ { address \longrightarrow addr, wheatherstaticalinformation \longrightarrow wheatherstatistic }

G A4(address, wheatherstaticalinformation)

Def \emptyset

has_opt false