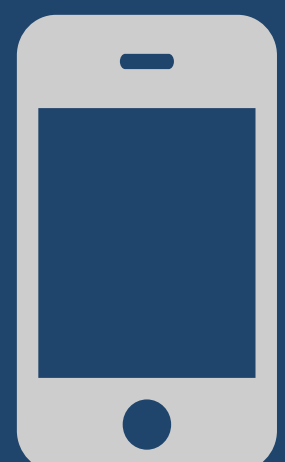


- MOTIVATION
- **Conjunctive** queries (AND operations) can unexpectedly lead to **empty result sets**
 - **OPTIONAL** queries can lead to extensively **massive** and intransparent result sets
 - **Preferences** model explicit wishes naturally over incomplete data sets
 - Some **preferences may depend** on other wishes to be fulfilled (as shown in the example)
 - Our previous work lacks this ability

- STRUCTURE
1. Isolate common types of **use cases** of inter-dependent preference models
 2. Develop a **syntax** that is easy to understand and as concise as possible
 3. Derive how these queries should be handled **semantically**
 4. Extend our existing **query rewriting** technique to accomodate for our modifications
 5. Discuss **performance** implications

- RESULTS
- Preferences deliver results somewhere between conjunctions and OPTIONAL patterns
 - Framework already usable via query rewriting
 - New operators solve modeling problems of our framework
 - Performance should stay the same or better

Preferences offer an intuitive way to retrieve complex data structures from incomplete knowledge graphs.



Take a picture to download the full paper

Examples

`?car is_a Car OPTIMAL (?car color ?color)`

Minimal example of our framework

```
SELECT ?car ?manufacturer ?region ?color
WHERE {
  ?car a :Car
  OPTIMAL (
    {
      ?car :manufacturer ?manufacturer .
      BIND(1 AS ?has_manufacturer)
    },
    {
      ?manufacturer :based_in ?region
      FILTER( bound(?has_manufacturer) )
    },
    {
      ?car :color ?color
    }
  )
}
```

Dependencies prior to our revision

```
SELECT ?car ?manufacturer ?country ?color
WHERE {
  ?car a :Car
  OPTIMAL(
    { ?car :made_by ?manufacturer }
    THEN {
      ?manufacturer :based_in ?country
    },
    { ?car :color ?color }
  )
}
```

Dependencies after our revision

 Till Affeldt, Stephan Mennicke,
Wolf-Tilo Balke

