U.PORTO
FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

JanusGraph

M.EIC - Non-Relational Databases 2022/2023

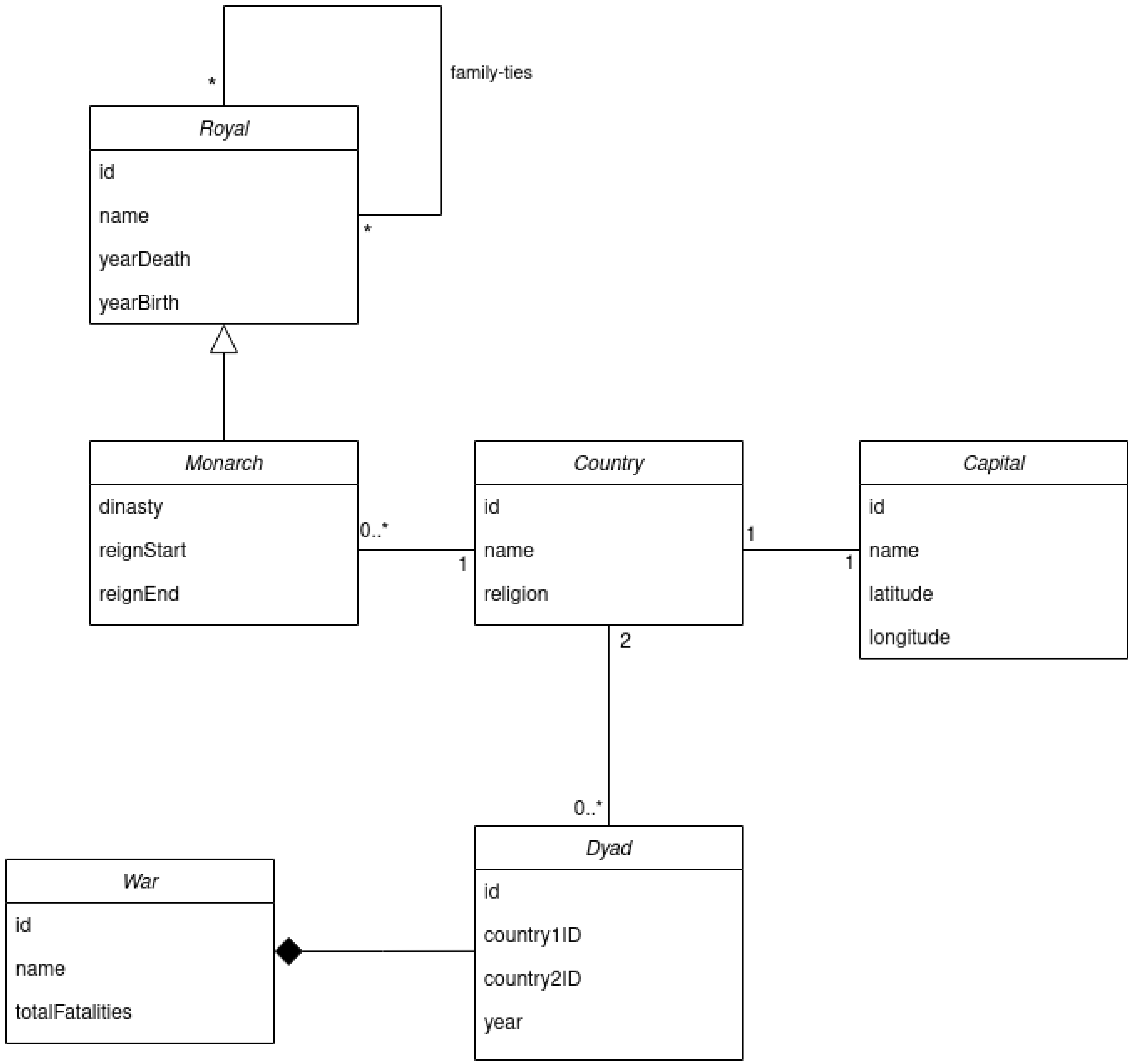
José Silva - up201904775@up.pt | Maria Carneiro - up201907726@up.pt | Sérgio Estêvão - up201905680@up.pt

Conceptual Data Model

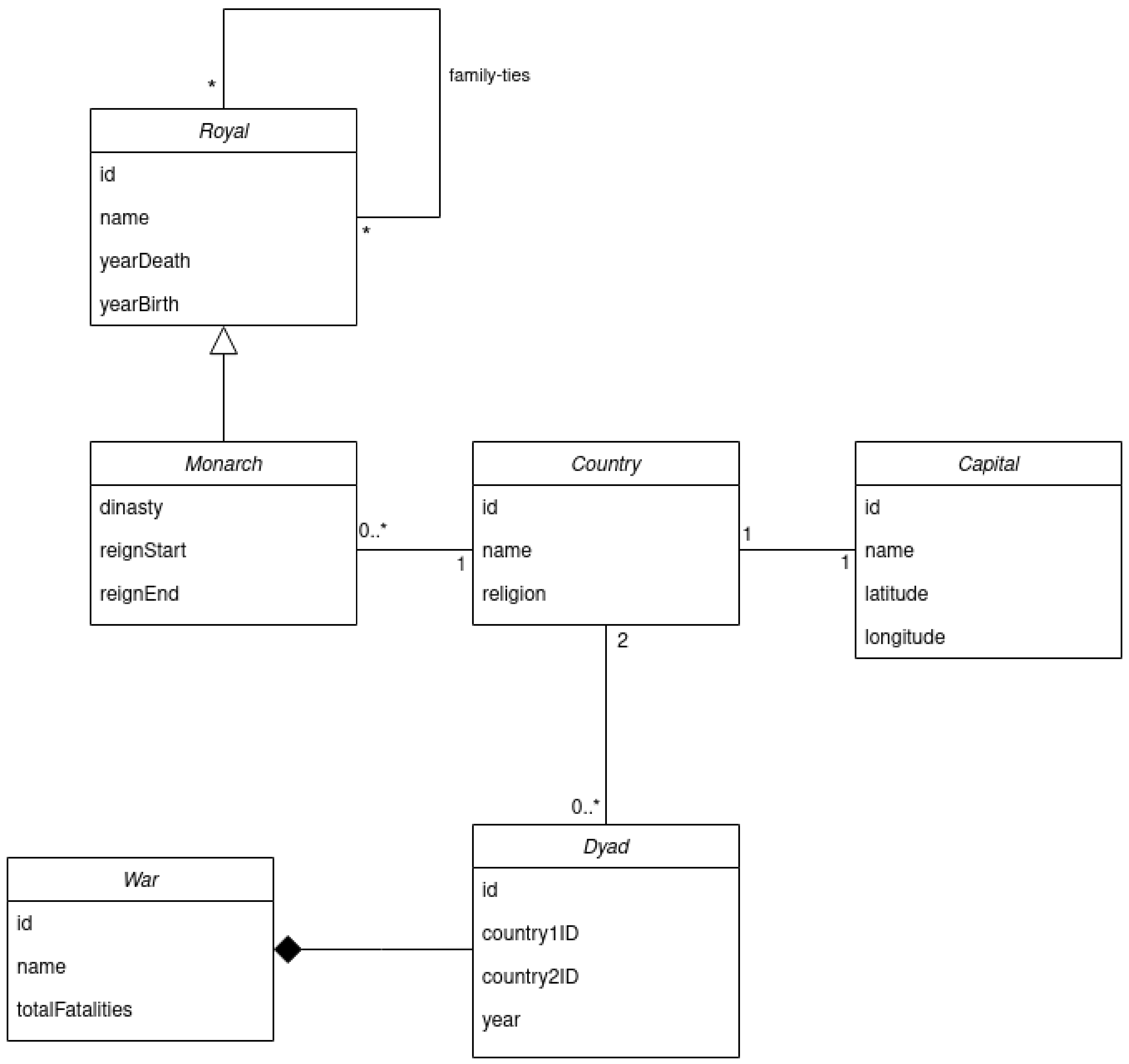
Dataset that links European royal kinship networks, monarchies, and wars to study the effect of family ties on conflict, from 1495–1918.

Family-ties can either represent marriages between royals or kinship relationships, since some married royals are blood-related.

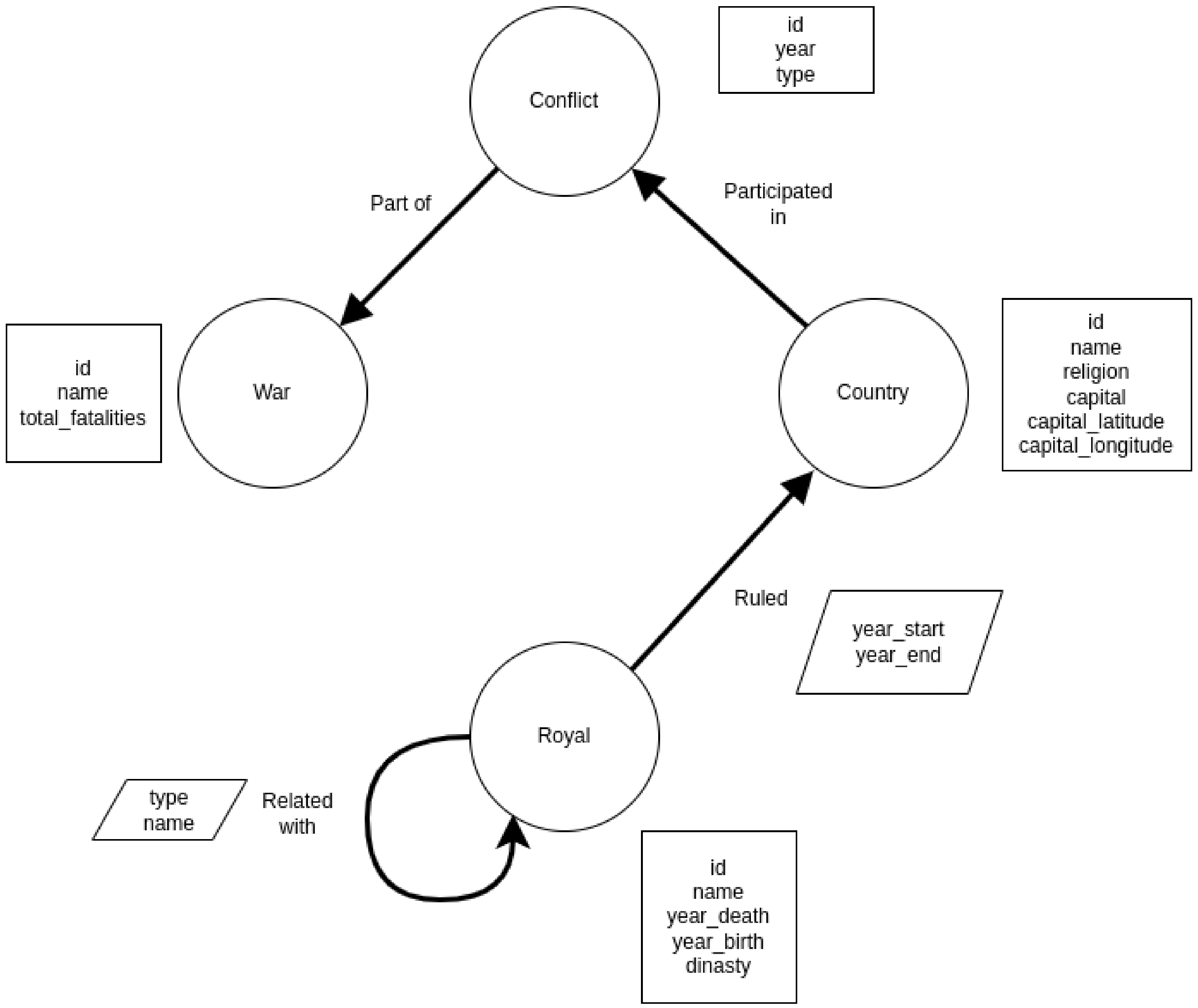
Dyads represent conflicts between two countries in a given war.



Conceptual Data Model



Graph Data Model

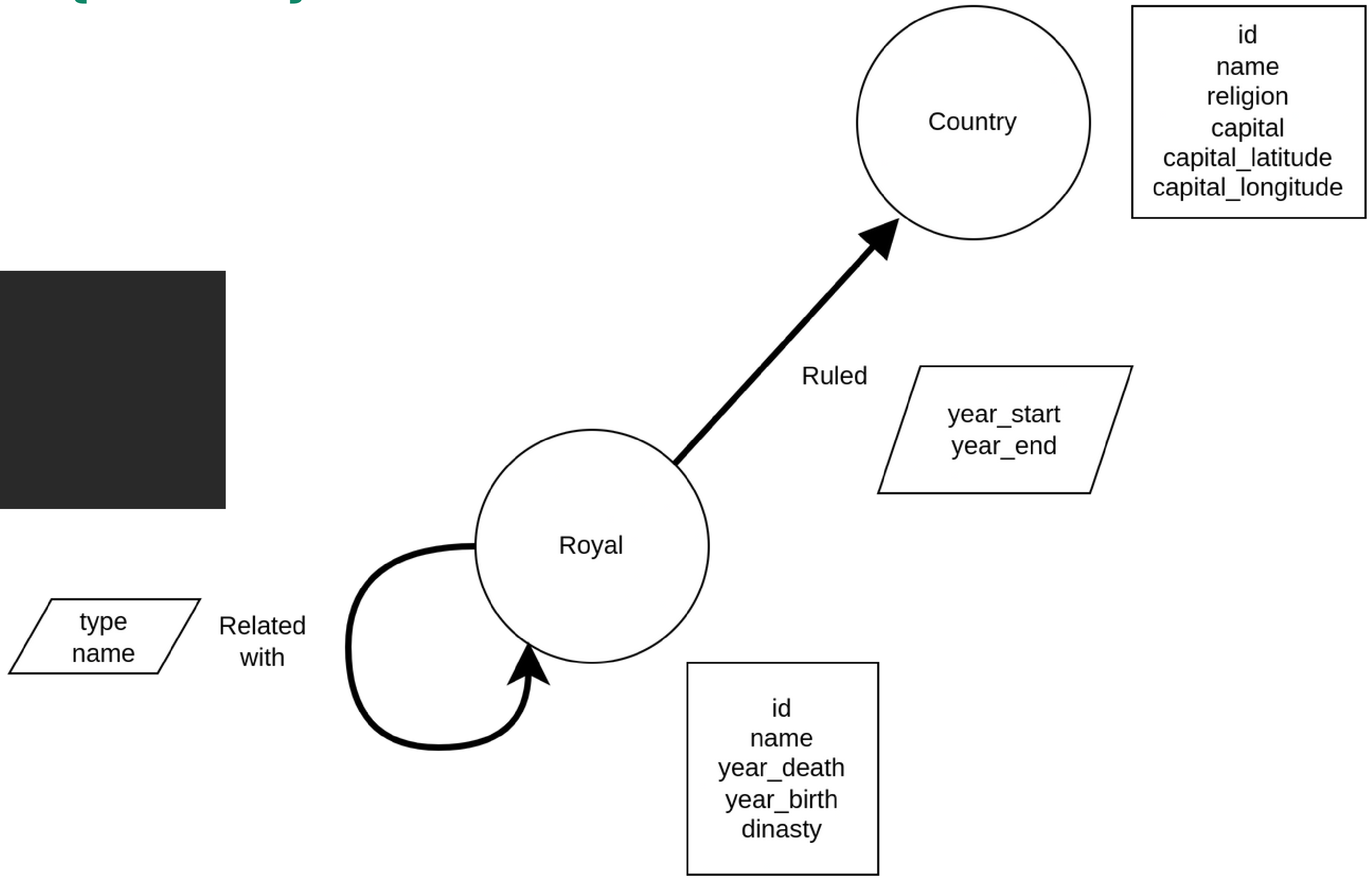


Proposed Query #1

I want to know all the royals that reigned over Portugal until {1650}.

Gremlin Query:

```
g.V().has('name', 'Portugal').in('ruled').
 has('year_end', lte(1650))
```

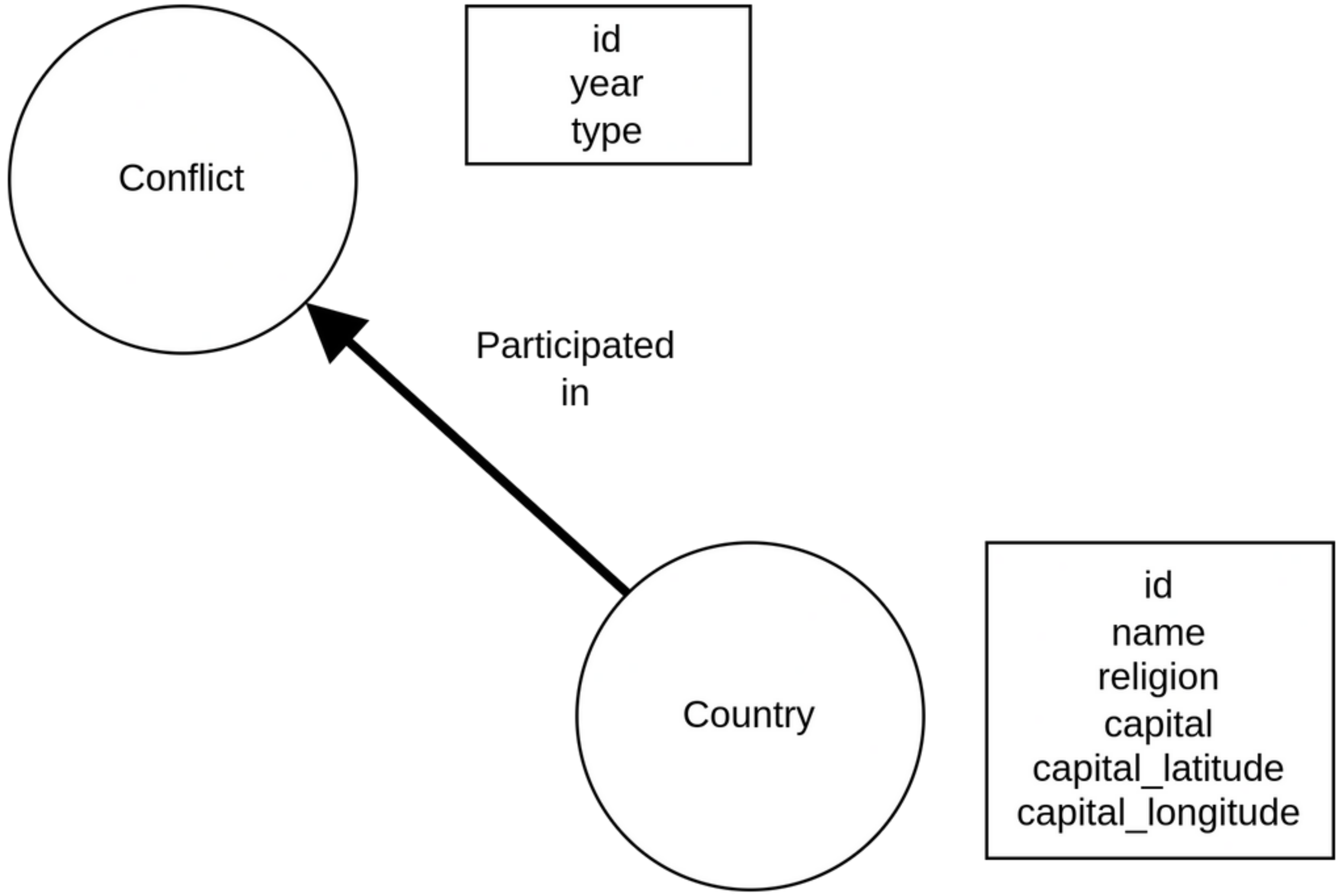


Proposed Query #2

I want to know the conflicts that happened between {England} and {France} between {1500} and {1600}.

Gremlin Query:

```
g.V().has('name', 'England').as('england').
  out('participated in').as('conflict').
  in('participated in').has('name', 'France').
  where('england', eq('it')).
  where('conflict', has('year').is(between(1500,1600))).
  select('conflict').by('name')
```



Conflict

id
year
type

Participated in

Country

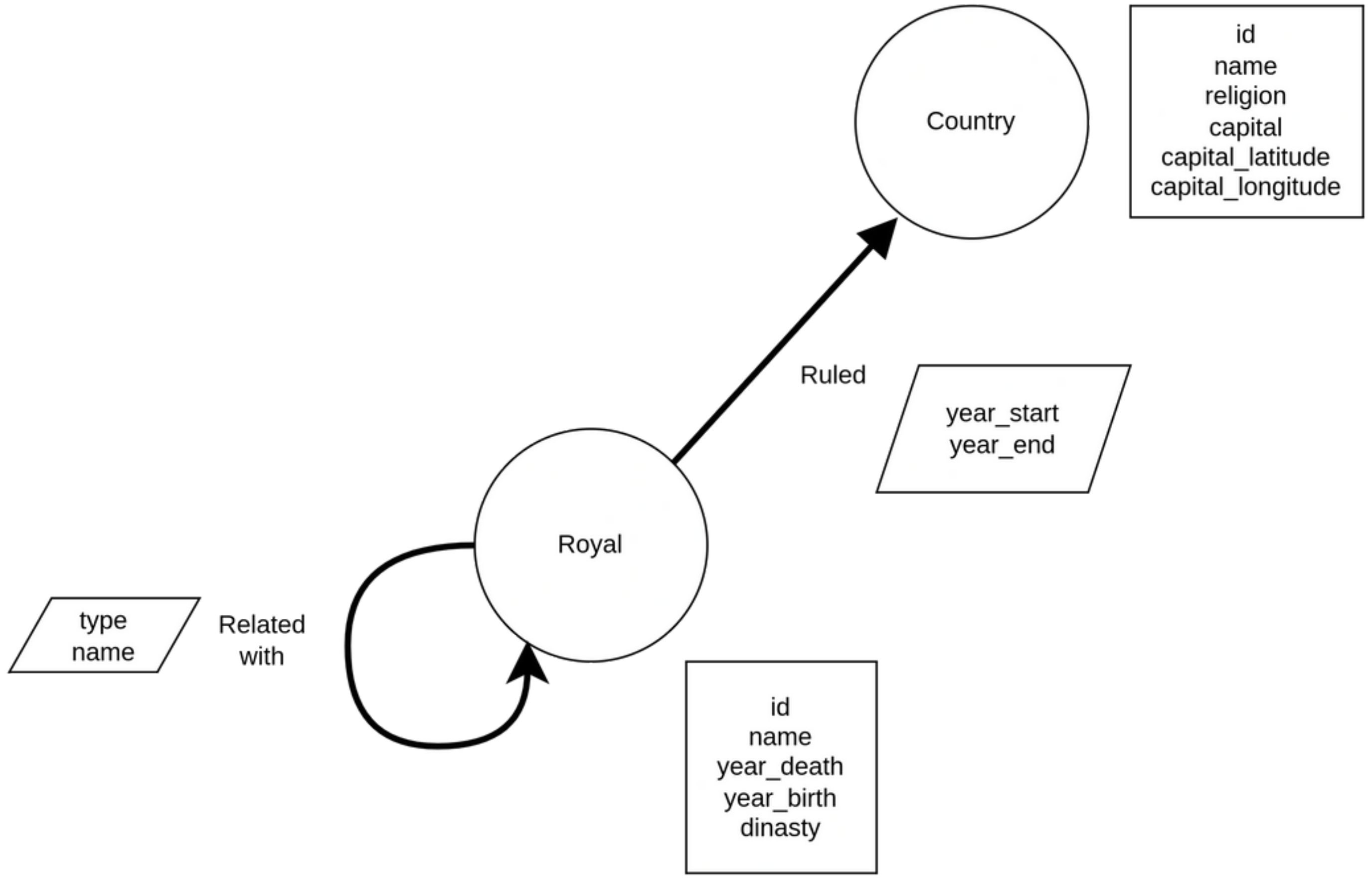
id
name
religion
capital
capital_latitude
capital_longitude

Proposed Query #3

I want to know all the descendants of {Catherine of Aragon} that reigned countries with capitals in a {100km} radius of {London}.

Gremlin Query:

```
g.V().has('name', 'Catherine of Aragon').repeat(out('related with')).emit().
 has('year_birth',gt(1500)).as('descendant').
 out('ruled').has('capital_latitude').
 has('capital_longitude').
 filter(
 geo.distance(
 values('capital_latitude','capital_longitude'),
 g.V().has('name','London').values('capital_latitude','capital_longitude')
).is(lte(100))).
 select('descendant').by('name')
```



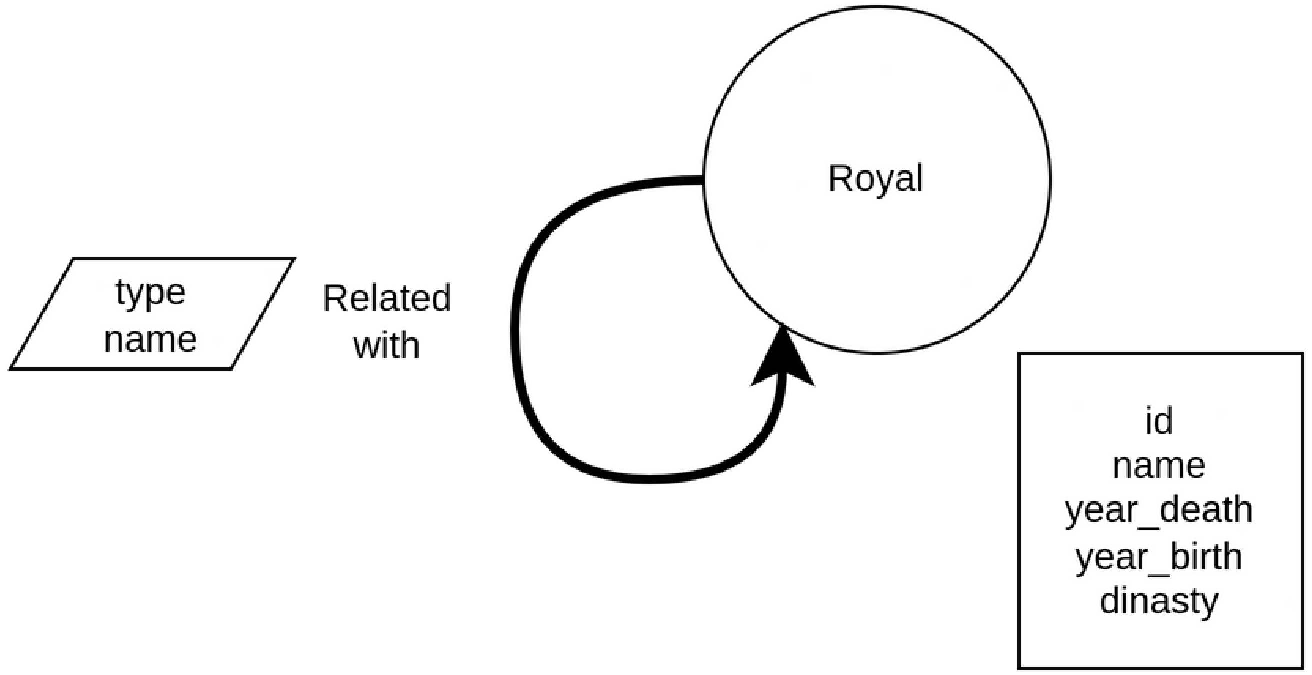
JanusGraph supports geo search

Proposed Query #4

I want to see the kinship distance between {Elizabeth of France} and {Philip V of Spain}.

Gremlin Query:

```
g.V().has('name', 'Elizabeth of France').repeat(out('related with')).emit().
  path().as('elizabethPath').by('name').
  V().has('name', 'Philip V of Spain').repeat(out('related with')).emit().
  path().as('philipPath').by('name').
  limit(1).
  select('elizabethPath','philipPath').
  map{
    def elizabethSet = it.get().get('elizabethPath').toSet();
    def philipSet = it.get().get('philipPath').toSet();
    return elizabethSet.intersect(philipSet).size();
  }
```



Questions?