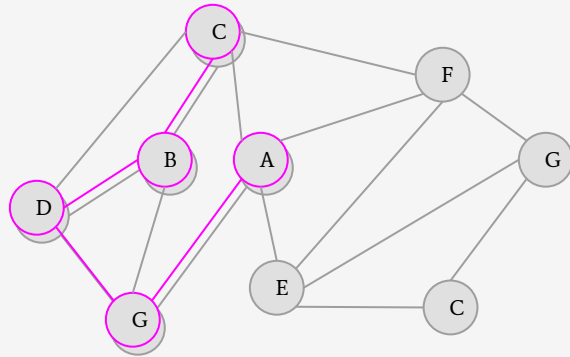


# Dotazovací jazyk SPARQL SELECT

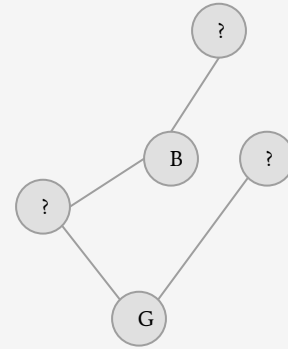


ISKM89 Organizace dat - sémantický web | podzim 2023  
Zuzana Nevěřilová | Centrum zpracování přirozeného jazyka

# SPARQL - graph pattern matching

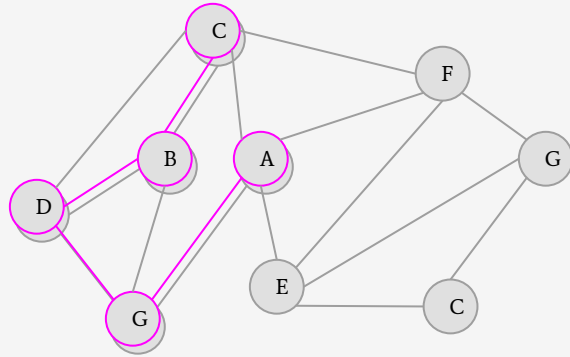


RDF graf



SPARQL query

# SPARQL - graph pattern matching



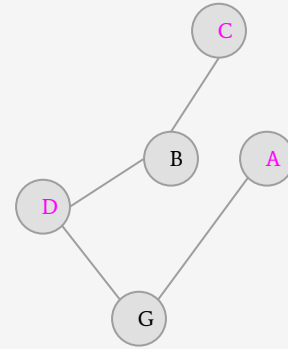
RDF graf

? = A

? = C

? = D

**binding**



SPARQL query

# Základní a pokročilé operátory

operátor	typ	výsledek
!A	xsd:boolean	xsd:boolean
+A	číslo	číslo
-A	číslo	číslo
BOUND(A)	proměnná	xsd:boolean
isURI(A)	RDF term	xsd:boolean
isBLANK(A)	RDF term	xsd:boolean
isLITERAL(A)	RDF term	xsd:boolean
STR(A)	litérál nebo URL	litérál
LANG(A)	litérál	litérál
DATATYPE(A)	litérál	URI

logické spojky && (AND) || (OR)

operátory s čísly, daty, řetězci = != < > <= >=

porovnávání = !=

aritmetické operátory + - / \* MAX(), MIN(), ...

REGEX(string, pattern, [flags])- regulární výrazy

sameTERM(A, B)

langMATCHES(A, B)

# Základní a pokročilé operátory

- FILTER a regulární výrazy
- OPTIONAL
- UNION
- negace
- property paths

# SPARQL SELECT - FILTER REGEX

Regulární výraz (REGEX) je zápis vzoru řetězce, který chceme hledat:

- "kočka" najde řetězec "kočka"
- "(Kk)očka" najde řetězec "kočka" nebo "Kočka"
- "(kočka|pes)" najde řetězec "kočka" nebo pes
- "\\bjen\\b" najde slovo "jen" (nenajde "nejen")
- "^ [0-9]" najde číslo na začátku
- "[0-9]\$" najde číslo na konci

Flags (modifikátory):

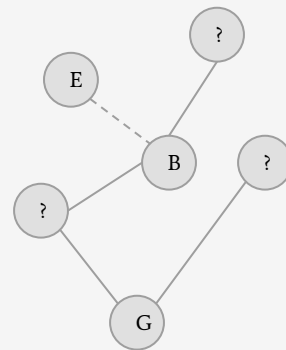
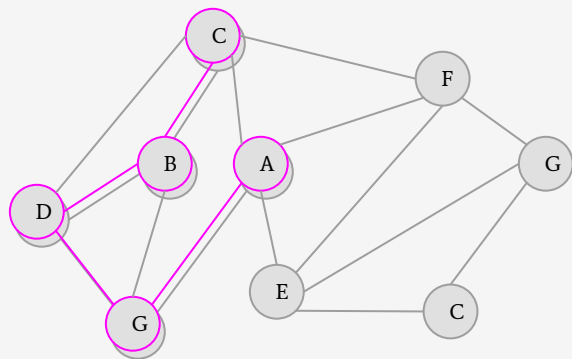
- i - case insensitive (nezáleží na malých a velkých písmenech)

# SPARQL SELECT - OPTIONAL

OPTIONAL přidává do grafu možnost

komplexita dotazu se zvyšuje

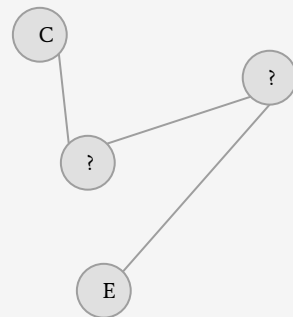
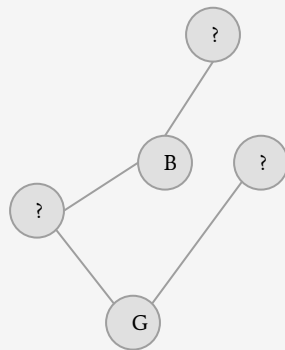
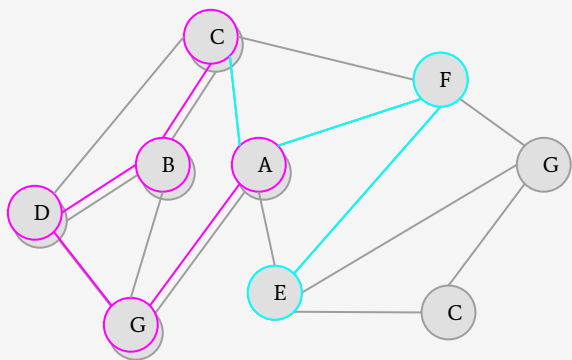
můžeme omezeně nahradit chybějící disjunkci (OR)



# SPARQL SELECT - UNION

UNION - skutečná disjunkce

{podgraf1} UNION {podgraf2}





# SPARQL SELECT - NEGACE

použití FILTER a !=, případně dalších operátorů

test, zda je OPTIONAL proměnná BOUND (např. !BOUND(?image))

- název nezačíná na K
- počet stran není větší než 100 (je menší nebo roven 100)
- ke knize neexistuje obrázek

FILTER NOT EXISTS {s p o} - odpovídá graf, který nemá trojici s p o

ještě obecněji, odpovídá graf, který nemá podgraf specifikovaný v {}

může být výpočetně náročné

# SPARQL SELECT WHERE SELECT (zanořování)

vnější SELECT se vypočítá po výpočtu vnitřního SELECTu (subquery)

```
SELECT ?countryLabel ?population (round(?population/?worldpopulation*1000)/10 AS ?percentage)
WHERE {
  ?country wdt:P31 wd:Q3624078;      # is a sovereign state
          wdt:P1082 ?population.

  {
    # subquery to determine ?worldpopulation
    SELECT (sum(?population) AS ?worldpopulation)
    WHERE {
      ?country wdt:P31 wd:Q3624078;      # is a sovereign state
              wdt:P1082 ?population.
    }
  }

  SERVICE wikibase:label {bd:serviceParam wikibase:language "[AUTO_LANGUAGE],en".}
}

ORDER BY desc(?population)
```

<https://en.wikibooks.org/wiki/SPARQL/Subqueries>

# SPARQL Property Paths

Standardně je uvnitř WHERE podgraf, kde jsou vlastnosti (properties) uvnitř trojice.

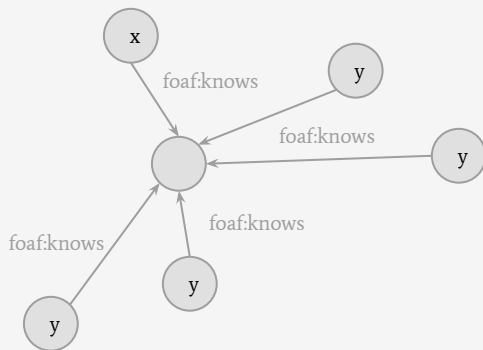
**Property path** je možná cesta mezi uzly A a B:

- **alternativy** { :book dc:title|rdfs:label ?book\_title .}
- **sekvence** { :alice foaf:knows/foaf:knows/foaf:name ?someone. }
- **inverzní relace** { ?someone foaf:mbox ?email. } = { ?email ^foaf:mbox ?someone. }
- **libovolná délka** { ?x foaf:knows+/foaf:name ?y }
- **negace** { ?x !rdf:type ?y }

# SPARQL Property Paths - speciality

inverze a sekvence - kdo jiný zná stejné lidi jako já?

```
{?x foaf:knows/^foaf:knows ?y. FILTER (?x != ?y) }
```



# SPARQL Property Paths - speciality

inverze a sekvence - jak se jmenují známí mých známých a jejich známých?

```
{?x foaf:knows+/foaf:name ?name}
```

+ nejméně jedna vlastnost

\* libovolná délka (i 0)

