



Databáze ◊ poznámky k přednášce

9. Hluboké dokumenty

verze z 25. listopadu 2024

Pro motivaci uvažujme kolečkové proměnnou `director` nad $\{_id, name, movie.\{title, year\}\}$ danou vlastností:

„Režisér `_id` se jmenuje `name`.“

kde `movie` je dáno vlastností:

„Režisér `_id` natočil v roce `movie.year` film `movie.title`.“

Hodnota proměnné je:

```
[
  {
    _id: 1,
    name: 'Andrei Tarkovsky',
    movie: [
      { title: 'Solaris', year: 1972 },
      { title: 'Stalker', year: 1979 }
    ]
  },
  {
    _id: 2,
    name: 'Francis Ford Coppola',
    movie: [
      { title: 'The Godfather', year: 1972 },
      { title: "Bram Stoker's Dracula", year: 1992 }
    ]
  }
]
```

1 Hluboké podmínky

Schéma S_2 je **podschéma** schématu S_1 , jestliže $Y(S_2) \subseteq Y(S_1)$ a pro každý atribut $y \in Y(S_2)$ platí, že $S_2(y)$ je podschéma $S_1(y)$. Například $\{name, movie.\{title\}\}$ je podschéma $\{_id, name, movie.\{title, year\}\}$.

Pokud je podmínka nad S_2 a S_2 je podschéma S_1 , pak je podmínka i nad S_1 . Například podmínka `name = "Francis Ford Coppola"` je nad schématem $\{name, movie.\{title\}\}$ a tedy i nad schématem $\{_id, name, movie.\{title, year\}\}$.

Vezměme kolekční atribut y , schéma S a podmínku θ nad S . Pak

$$y.(\theta)$$

je podmínka nad $\{y.S\}$, kterou dokument t splňuje, jestliže existuje v kolekci $t(y)$ dokument, který splňuje θ .

Například podmínku

$$\text{movie}.\text{(year = 1979)}$$

dokument t splňuje, jestliže v kolekci $t(\text{movie})$ existuje dokument t' takový, že $t'(\text{year})$ je 1979. Dokument:

```
{
  _id: 1,
  name: 'Andrei Tarkovsky',
  movie: [
    { title: 'Solaris', year: 1972 },
    { title: 'Stalker', year: 1979 }
  ]
}
```

podmínku splňuje.

Podmínku $y.(\theta)$ zapíšeme řetězcem:

```
{ "y": { "$elemMatch":  $\theta$  } }
```

Například:

```
> db.director.find({ movie: { $elemMatch: { year: 1979 } } })
```

```
[
  {
    _id: 1,
    name: 'Andrei Tarkovsky',
    movie: [
      { title: 'Solaris', year: 1972 },
      { title: 'Stalker', year: 1979 }
    ]
  }
]
```

Charakteristická vlastnost předchozí kolekce je: „Režisér $_id$, který se jmenuje $name$, natočil film v roce 1979.“ Charakteristická vlastnost kolekce $movie$ zůstává stejná: „Režisér $_id$ natočil v roce $movie.year$ film $movie.title$.“

2 Hluboké projekce

Projekce kolekce C nad S_1 na podschéma S_2 je kolekce:

$$\pi_{S_2}(C) = \{t(S_2) \mid t \in C\}$$

kde $t(S_2)$ je **projekce dokumentu** t nad S_1 na S_2 , která je určena následovně.

$$t(S_2) = \{\langle y, d \rangle \in t \mid y \text{ je atomický atribut}\} \\ \cup \\ \{\langle y, \pi_{S'}(d) \rangle \mid \langle y, d \rangle \in t \text{ a } \langle y, S' \rangle \in S_2 \text{ a } y \text{ je kolekční atribut}\}.$$

Například projekce dokumentu:

```
{
  _id: 1,
  name: 'Andrei Tarkovsky',
  movie: [
    { title: 'Solaris', year: 1972 },
    { title: 'Stalker', year: 1979 }
  ]
}
```

na schéma $\{\text{name, movie.\{title\}}\}$ je dokument:

```
{
  name: 'Andrei Tarkovsky',
  movie: [
    { title: 'Solaris' },
    { title: 'Stalker' }
  ]
}
```

Označme $Y' = \{y_1, \dots, y_m\} = \{y \in Y(S_2) \mid S_1(y) = S_2(y)\}$ a $\{x_1, \dots, x_n\} = Y(S_2) - Y'$. Pak schéma S_2 zapíšeme řetězcem:

```
{
  y_1: 1,
  :
  y_m: 1,
  x_1: S(x_1),
  :
  x_n: S(x_n)
}
```

Pokud $_id \notin Y(S_2)$ a schéma S_2 není zanořené v jiném schématu, pak S_2 zapíšeme řetězcem:

```
{
  _id: 0,
  y1: 1,
  :
  ym: 1,
  x1: S(x1),
  :
  xn: S(xn)
}
```

Například schéma $S_2 = \{\text{name, movie.\{title\}}\}$ pro projekci z $S_1 = \{_id, \text{name, movie.\{title, year\}}\}$ zapíšeme:

```
{
  _id: 0,
  title: 1,
  movie: {
    title: 1
  }
}
```

Vezměme kolekční proměnnou *collection* a podmínku θ nad S_1 a podschéma S_2 schématu S_1 . Pak hodnotou kolekčního výrazu:

```
db.collection.find( $\theta$ ,  $S_2$ )
```

je kolekce:

$$\pi_{S_2}(\sigma_{\theta}(C_{collection})).$$

Například:

```
> db.director.find({}, { _id: 0, name: 1, movie: { title: 1 } })
```

```
[
  {
    name: 'Andrei Tarkovsky',
    movie: [ { title: 'Solaris' }, { title: 'Stalker' } ]
  },
  {
    name: 'Francis Ford Coppola',
    movie: [ { title: 'The Godfather' }, { title: "Bram Stoker's Dracula" } ]
  }
]
```

je kolekce dána vlastností: „Řetězec *name* je jméno režiséra.“ a kolekční atribut *movie* vlastností: „Režisér se jménem *name* natočil film *movie.title*.“

3 Hluboké změny

Vezměme dokument t nad S , $y \in Y(S)$ a hodnotu v . Pak **nastavení hodnoty atributu** y v dokumentu t na v je dokument

$$t(S') \cup \{\langle y, v \rangle\}$$

označený $t(y/v)$, kde $S' = \{\langle y', S'' \rangle \in S \mid y' \neq y\}$.

Například pro $t = \{\langle \text{title}, "Solaris" \rangle, \langle \text{year}, 1972 \rangle\}$ je

$$t(\text{title}/"The Godfather") = \{\langle \text{title}, "The Godfather" \rangle, \langle \text{year}, 1972 \rangle\}.$$

Příkaz

```
db.collection.updateMany( $\theta_1$ , {
  "$pull": {
    "y":  $\theta_2$ 
  }
})
```

nastaví hodnotu proměnné *collection* na

$$\sigma_{-\theta_1}(C) \cup \{t(y/\sigma_{-\theta_2}(t(y))) \mid t \in \sigma_{\theta_1}(C)\},$$

kde C je původní hodnota proměnné *collection*. Například vezměme proměnnou:

```
> db.director.find()

[
  {
    _id: 1,
    name: 'Andrei Tarkovsky',
    movie: [
      { title: 'Solaris', year: 1972 },
      { title: 'Stalker', year: 1979 }
    ]
  },
  {
    _id: 2,
    name: 'Francis Ford Coppola',
    movie: [
      { title: 'The Godfather', year: 1972 },
      { title: "Bram Stoker's Dracula", year: 1992 }
    ]
  }
]
```

Pak příkaz:

```
> db.director.updateMany({}, {
  $pull: {
    movie: {
      year: 1972
    }
  }
})
```

odstraní všechny filmy z roku 1972. Tedy:

```
> db.director.find()

[
  {
    _id: 1,
    name: 'Andrei Tarkovsky',
    movie: [ { title: 'Stalker', year: 1979 } ]
  },
  {
    _id: 2,
    name: 'Francis Ford Coppola',
    movie: [ { title: "Bram Stoker's Dracula", year: 1992 } ]
  }
]
```

Příkaz:

```
db.collection.updateMany( $\theta$ ,
  { "$push": {
    "y": {
      "$each":  $C'$ 
    }
  }
})
```

nastaví hodnotu proměnné *collection* na

$$\sigma_{-\theta}(C) \cup \{t(y/(t(y) \cup C')) \mid t \in \sigma_{\theta}(C)\}$$

kde C je původní hodnota proměnné *collection*. Musí platit, že $t(y)$ a C' jsou disjunktní pro každý dokument $t \in \sigma_{\theta}(C)$. Například příkaz:

```
> db.director.updateMany(
  { _id: 2 },
```

```

{ $push: {
  movie: {
    $each: [
      { title: 'Megalopolis', year: 2024 }
    ]
  }
}
})

```

Přidá film Megalopolis k režisérovi Francis Ford Coppola:

```
> db.director.find()
```

```

[
  {
    _id: 1,
    name: 'Andrei Tarkovsky',
    movie: [ { title: 'Stalker', year: 1979 } ]
  },
  {
    _id: 2,
    name: 'Francis Ford Coppola',
    movie: [
      { title: "Bram Stoker's Dracula", year: 1992 },
      { title: 'Megalopolis', year: 2024 }
    ]
  }
]

```

Otázky a úkoly na cvičení

- Je dáno schéma `{_id, title, actor.{name, born}}` určete, která následující schémata jsou jeho podschémata.
 - `{_id, actor.{name}}`
 - `{actor.{born}, title}`
 - `{title, _id}`
 - `{actor.{title}, name}`
 - `{title}`
 - `{_id, name}`
- Je dána proměnná `movie` nad `{_id, title, actor.{name, born}}` určená vlastností „Film `_id` se jmenuje `title`.“, kde kolečkový atribut `actor` je určen vlastností „Ve filmu `title` hraje herec `actor.name` narozený roku `actor.born`“. Proměnná má hodnotu:

```
[ {
  _id: 1,
  title: 'Megalopolis',
  actor: [
    { name: 'Adam Driver', born: 1983 },
    { name: 'Laurence Fishburne', born: 1961 }
  ]
}, {
  _id: 2,
  title: 'Ferrari',
  actor: [
    { name: 'Adam Driver', born: 1983 },
    { name: 'Penélope Cruz', born: 1974 }
  ]
} ]
```

Proměnnou si vytvořte v MongoDB. Určete charakteristické vlastnosti a hodnoty následujících restrikcí, kde C_{movie} je hodnota proměnné `movie`.

- $\sigma_{\text{actor}.\{\text{born}=1974\}}(C_{\text{movie}})$
- $\sigma_{(\text{title}=\text{"Megalopolis"})\wedge\text{actor}.\{\text{name}=\text{"Adam Driver"}\}}(C_{\text{movie}})$
- $\sigma_{\text{actor}.\{(\text{name}=\text{"Laurence Fishburne"})\vee(\text{name}=\text{"Penélope Cruz"})\}}(C_{\text{movie}})$
- $\sigma_{\neg\text{actor}.\{\text{name}=\text{"Laurence Fishburne"}\}}(C_{\text{movie}})$
- $\sigma_{\text{actor}.\{\text{name}=\text{"Adam Driver"}\}\wedge\text{actor}.\{\text{name}=\text{"Penélope Cruz"}\}}(C_{\text{movie}})$

Restrikce zapište kolekčními výrazy a ověřte, že vámi vypočítané hodnoty se shodují s hodnotami vypočítanými v MongoDB.

- Určete restrikce, které mají následující charakteristické vlastnosti. Charakteristická vlastnost kolekčního atributu `actor` zůstává stále stejná.
 - Ve filmu `_id` s názvem `title` hraje herec jménem Laurence Fishburne.
 - Ve filmu `_id` s názvem `title` hraje herec narozený roku 1974 nebo 1975.
 - Ve filmu `_id` s názvem `title` hrají současně herci Laurence Fishburne a Penélope Cruz.
 - Ve filmu `_id` rovným 1 s názvem `title` hraje herec narozený roku 1983.

Určete hodnoty restrikcí. Restrikce zapište kolekčními výrazy a vaše výpočty ověřte v MongoDB.

- Určete hodnotu a charakteristickou vlastnost projekce

$$\pi_{\{\text{title}, \text{actor}.\{\text{born}\}\}}(C_{\text{movie}}).$$

Projekci zapište kolekčním výrazem a jeho hodnotu ověřte v MongoDB.

5. Zapište pomocí projekce a restrikce kolekci s charakteristickou vlastností: „Ve filmu s názvem `title` hraje herec narozený roku 1983.“ a kolekčním atributem `actor` s charakteristickou vlastností „Ve filmu s názvem `title` hraje herec `actor.name`“. Získanou kolekci zapište kolekčním výrazem a získejte její hodnotu v MongoDB.
6. Nastavte rok narození Adama Drivera na chybnou hodnotu 1974 a pak chybu opravte.
7. Přidejte herce Giancarlo Esposito narozeného v roce 1958, který hrál ve filmu Megalopolis.
8. Odstraňte všechny herce narozené v roce 1983. Odstraněné herce následně přidejte zpět.