

8. Mělké dokumenty

verze z 19. listopadu 2024

1 Podmínky

Výrazy, u kterých můžeme určit, zda je dokument **splňuje**, nazýváme **podmínky**. Říkáme, že podmínka je **nad schématem** S , jestliže lze určit pro každý dokument nad S , zda ji splňuje.

Pro atomický atribut y a $d \in D_y$ je

$$y = d$$

podmínka nad $\{y\}$. Dokument t splňuje podmínku, jestliže $t(y) = d$. Například `title = "Gladiator"` je podmínka nad `{title}` a dokument:

```
{ title: 'Gladiator' }
```

ji splňuje.

Jestliže je podmínka nad S_1 a $S_1 \subseteq S_2$, pak je tato podmínka i nad S_2 . Například `title = "Gladiator"` je podmínka i nad `{title, year}` a dokument:

```
{ title: 'Gladiator', year: 1992 }
```

splňuje podmínku `title = "Gladiator"`.

Podmínky také zapisujeme ve formátu JSON. Podmínku $y = d$ zapíšeme:

```
{
  "y": {
    "$eq": d
  }
}
```

Například podmínku `title = "Gladiator"` zapíšeme řetězcem:

```
{
  "title": {
    "$eq": "Gladiator"
  }
}
```

Využitím stručnějšího zápisu můžeme stejnou podmínku zadat:

```
{
  title: {
    $eq: 'Gladiator'
  }
}
```

Podmínce θ přiřadíme výrokovou formu $V_\theta(t)$ rovnou „ t splňuje θ “. Například `title = "Gladiator"` má přiřazenou výrokovou formu $V_{\text{title}=\text{"Gladiator"}}(t)$ rovnou „ $t(\text{title})$ je rovno „Gladiator““, což můžeme zkrátit na „ title je „Gladiator““.

2 Restrikce

Vezměme kolekci C a podmínku θ nad S . Kolekce

$$\sigma_\theta(C) = \{t \in C \mid t \text{ splňuje } \theta\}$$

se nazývá **restrikcí** kolekce C podle θ . Například pro kolekci C :

```
[
  { _id: 1, title: 'Gladiator', year: 1992 },
  { _id: 2, title: 'Gladiator', year: 2000 },
  { _id: 3, title: 'X-Men', year: 2000 },
  { _id: 4, title: 'The Matrix', year: 1999 }
]
```

je $\sigma_{\text{title}=\text{"Gladiator"}}(C)$ kolekce:

```
[
  { _id: 1, title: 'Gladiator', year: 1992 },
  { _id: 2, title: 'Gladiator', year: 2000 }
]
```

Pokud C má charakteristickou vlastnost $V_C(t)$ a podmínka θ má výrokovou formu $V_\theta(t)$, pak restrikce $\sigma_\theta(C)$ má charakteristickou vlastnost „ $V_C(t)$ a $V_\theta(t)$ “. Například vezměme kolekci C danou: „Film s názvem `title` byl vydán roku `year`.“, pak kolekce $\sigma_{\text{title}=\text{"Gladiator"}}(C)$ je dána vlastností: „Film s názvem `title` rovnému `Gladiator` byl vydán roku `year`.“

Hodnotu kolekční proměnné *collection* budeme značit $C_{\text{collection}}$. Restrikci $\sigma_\theta(C_{\text{collection}})$ hodnoty proměnné *collection* podle θ zapíšeme kolekčním výrazem:

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

Vezměme například kolekční proměnnou `movie` danou vlastností: „Film `_id` s názvem `title` byl vydán roku `year`.“ s hodnotou:

```
[
  { _id: 1, title: 'Gladiator', year: 1992 },
  { _id: 2, title: 'Gladiator', year: 2000 },
  { _id: 3, title: 'X-Men', year: 2000 },
  { _id: 4, title: 'The Matrix', year: 1999 }
]
```

Pak výraz:

```
db.movie.find({ title: { $eq: "Gladiator" } })
```

má hodnotu:

$$\theta_{\text{title}=\text{"Gladiator"}}(C_{\text{movie}})$$

restrikce C_{movie} podle `title = "Gladiator"`. Tedy:

```
> db.movie.find({ title: { $eq: "Gladiator" } })
```

```
[
  { _id: 1, title: 'Gladiator', year: 1992 },
  { _id: 2, title: 'Gladiator', year: 2000 }
]
```

Kolekce má charakteristickou vlastnost: „Film `_id` s názvem `title` rovný `Gladiator` byl vydán roku `year`.“

Řetězec:

```
{
  "y": d
}
```

je zkratkou za podmínku:

```
{
  "y": {
    "$eq": d
  }
}
```

Například:

```
> db.movie.find({ title: "Gladiator" })

[
  { _id: 1, title: 'Gladiator', year: 1992 },
  { _id: 2, title: 'Gladiator', year: 2000 }
]
```

Pro atomický atribut y a hodnotu d z domény atributu y je

$$y \neq d$$

podmínka nad $\{y\}$. Dokument t splňuje podmínku, jestliže $t(y) \neq d$. Zapisujeme ji řetězcem:

```
{
  "y": {
    "$ne": d
  }
}
```

Například restrikce:

```
> db.movie.find({ year: { $ne: 2000 } })

[
  { _id: 1, title: 'Gladiator', year: 1992 },
  { _id: 4, title: 'The Matrix', year: 1999 }
]
```

je dána: „Film $_id$ s názvem $title$ byl vydán roku $year$ různého od 2000.“

Vezměme podmínky $\theta_1, \dots, \theta_n$ nad S , kde $0 < n$, pak

$$\theta_1 \wedge \dots \wedge \theta_n$$

je podmínka nad S , kterou dokument splňuje, jestliže splňuje všechny podmínky $\theta_1, \dots, \theta_n$. Podmínku zapíšeme řetězcem:

```
{
  "$and": [
    \theta_1,
    \vdots
    \theta_n
  ]
}
```

Například podmínku $title = "Gladiator" \wedge year = 2000$ zapíšeme řetězcem:

```

{
  $and: [{
    title: {
      $eq: "Gladiator"
    }
  }, {
    year: {
      $eq: 2000
    }
  }]
}

```

který můžeme zkrátit na:

```

{
  $and: [{
    title: "Gladiator"
  }, {
    year: 2000
  }]
}

```

Například máme:

```

> db.movie.find({
  $and: [{
    title: "Gladiator"
  }, {
    year: 2000
  }]
})

```

```
[ { _id: 2, title: 'Gladiator', year: 2000 } ]
```

Kolekce je dána vlastností: „Film `_id` s názvem `title` rovnému `Gladiator` byl vydán roku `year` rovnému `2000`.“

Řetězec:

```

{
  "y1": expr1,
  :
  "yn": exprn
}

```

kde atributy y_1, \dots, y_n ($0 < n$) jsou po dvou různé, je zkratkou za podmínku:

```
{
  "$and": [
    { "y1": expr1 },
    :
    { "yn": exprn }
  ]
}
```

Například:

```
{
  title: "Gladiator",
  year: 2000
}
```

je zkratkou za:

```
{
  $and: [{
    title: "Gladiator"
  }, {
    year: 2000
  }]
}
```

Vezměme podmínky $\theta_1, \dots, \theta_n$ nad S , kde $0 < n$, pak

$$\theta_1 \vee \dots \vee \theta_n$$

je podmínka nad S , kterou dokument splňuje, jestliže splňuje aspoň jednu podmínku $\theta_1, \dots, \theta_n$. Podmínku zapíšeme řetězcem:

```
{
  "$or": [
     $\theta_1$ ,
    :
     $\theta_n$ 
  ]
}
```

Například podmínku `title = "Gladiator" \vee year = 2000` zapíšeme řetězcem:

```
{
  $or: [{
```

```

    title: "Gladiator"
  }, {
    year: 2000
  }
]
}

```

Restrikce:

```

> db.movie.find({
  $or: [{
    title: "Gladiator"
  }, {
    year: 2000
  }]
})

```

```

[
  { _id: 1, title: 'Gladiator', year: 1992 },
  { _id: 2, title: 'Gladiator', year: 2000 },
  { _id: 3, title: 'X-Men', year: 2000 }
]

```

je dána: „Film `_id` s názvem `title` byl vydán roku `year` a `title` je Gladiator nebo `year` je 2000.“

Pro podmínku θ nad S je

$$\neg\theta$$

podmínka nad S , kterou dokument splňuje, jestliže nesplňuje θ . Například $\neg(\text{title} = \text{"Gladiator"} \vee \text{year} = 2000)$ je podmínka nad $\{\text{title}, \text{year}\}$, kterou dokument splňuje, právě když neplatí, že `title` je Gladiator nebo `year` je 2000.

Dvě podmínky θ_1 a θ_2 nad S nazveme **ekvivalentní**, jestliže pro každý dokument t nad S platí, že t splňuje θ_1 , právě když t splňuje θ_2 . Například $\neg \text{year} = 2000$ je ekvivalentní s $\text{year} \neq 2000$. Ekvivalenci podmínek značíme $\theta_1 \equiv \theta_2$.

Negaci podmínky nemůžeme přímo zapsat řetězcem. Negaci můžeme z podmínky odstranit použitím De Morganových zákonů:

$$\begin{aligned} \neg(\theta_1 \wedge \dots \wedge \theta_n) &\equiv \neg\theta_1 \vee \dots \vee \neg\theta_n, \\ \neg(\theta_1 \vee \dots \vee \theta_n) &\equiv \neg\theta_1 \wedge \dots \wedge \neg\theta_n \end{aligned}$$

a vztahu negace rovnosti a nerovnosti:

$$\begin{aligned} \neg(y = d) &\equiv y \neq d, \\ \neg(y \neq d) &\equiv y = d. \end{aligned}$$

Například podmínku $\neg(\text{title} = \text{"Gladiator"} \vee \text{year} = 2000)$ upravíme:

$$\begin{aligned} \neg(\text{title} = \text{"Gladiator"} \vee \text{year} = 2000) &\equiv \neg\text{title} = \text{"Gladiator"} \wedge \neg\text{year} = 2000 \\ &\equiv \text{title} \neq \text{"Gladiator"} \wedge \text{year} \neq 2000 \end{aligned}$$

a zapíšeme:

```
{
  title: {
    $ne: "Gladiator"
  },
  year: {
    $ne: 2000
  }
}
```

Restrikce:

```
> db.movie.find({
  title: {
    $ne: "Gladiator"
  },
  year: {
    $ne: 2000
  }
})
```

```
[ { _id: 4, title: 'The Matrix', year: 1999 } ]
```

je dána: „Film `_id` s názvem `title` různým od `Gladiator` byl vydán roku `year` různého od 2000.“

Podmínku nad \emptyset , která je splněna v každém dokumentu, označíme 1 a zapisujeme řetězcem `{}`. Pro každou kolekci C máme, že $\sigma_1(C) = C$.

3 Projekce

Vezměme dokument t nad S_1 a $S_2 \subseteq S_1$. Pak **projekcí dokumentu** t na S_2 rozumíme dokument

$$t(S_2) = \{\langle y, v \rangle \in t \mid y \in Y(S_2)\}.$$

Například pro dokument t nad `{_id, title, year}`:

```
{ _id: 1, title: 'Gladiator', year: 1992 }
```

je $t(\{_id, title\})$ rovno:

```
{ _id: 1, title: 'Gladiator' }
```

Pro kolekci C nad S_1 je

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

kolekce nad S_2 **projekcí kolekce** C na S_2 . Například projekce kolekce C_{movie} :


```
[
  { _id: 1, title: 'Gladiator', year: 1992 },
  { _id: 2, title: 'Gladiator', year: 2000 },
  { _id: 3, title: 'X-Men', year: 2000 },
  { _id: 4, title: 'The Matrix', year: 1999 }
]
```

na `{title}` je kolekce:

```
[
  { title: 'Gladiator' },
  { title: 'Gladiator' },
  { title: 'X-Men' },
  { title: 'The Matrix' }
]
```

Všimněme si, že předchozí text zapisuje tříprvkovou kolekci.

Vezměme dokumenty t_1 nad S_1 a t_2 nad S_2 , kde $Y(S_1)$ a $Y(S_2)$ jsou disjunktní. Pak dokument $t_1 \cup t_2$ nad $S_1 \cup S_2$ je **spojení dokumentů** t_1 a t_2 a značíme jej $t_1 t_2$. Například spojení $\{\langle \text{title}, "Gladiator" \rangle\}$ a $\{\langle _id, 1 \rangle, \langle \text{year}, 1992 \rangle\}$ je dokument $\{\langle _id, 1 \rangle, \langle \text{title}, "Gladiator" \rangle, \langle \text{year}, 1992 \rangle\}$.

Nechť $V_C(t_1)$ je charakteristická vlastnost kolekce C nad S_1 , pak pro $S_2 \subseteq S_1$ má projekce $\pi_{S_2}(C)$ charakteristickou vlastnost $V(t_2)$ danou: „Existuje $t_3 \in \text{Doc}(S_1 - S_2)$ takové, že $V_C(t_2 t_3)$ “. Například vezměme kolekci C_{movie} danou vlastností „Film $_id$ s názvem title byl vydán roku year .“ Pak $\pi_{\{\text{title}\}}(C_{\text{movie}})$ má charakteristickou vlastnost $V(t_2)$:

„Existuje $t_3 \in \text{Doc}(\{_id, \text{year}\})$ tak, že film $t_3(\text{id})$ s názvem $t_2(\text{title})$ byl vydán roku $t_3(\text{year})$.“

Vlastnost můžeme zapsat bez proměnných t_2, t_3 :

„Existují $_id$ a year takové, že film id s názvem title byl vydán roku year .“

Což můžeme stručně zapsat jako: „Existuje film s názvem title .“ nebo ještě stručněji: „ title je název filmu.“

Pokud projikujeme na schéma S , kde $Y(S) = \{y_1, \dots, y_n\}$ obsahuje $_id$, tak S zapíšeme řetězcem:

```
{
  y1: 1,
  ⋮
  yn: 1
}
```

jinak S zapíšeme řetězcem:

```
{
  _id: 0,
  y1: 1,
  :
  yn: 1
}
```

Například schéma `{title}` zapíšeme řetězcem `{ _id: 0, title: 1 }`.

Vezměme kolekční proměnnou `collection` nad S_1 s hodnotou $C_{collection}$, pak kolekční výraz:

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

kde θ je podmínka nad S_1 a $S_2 \subseteq S_1$, má hodnotu

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

kterou je kolekce nad S_2 . Například:

```
> db.movie.find({ year: 2000 }, { _id: 0, title: 1})
```

```
[ { title: 'Gladiator' }, { title: 'X-Men' } ]
```

Kolekce je dána vlastností „V roce 2000 byl natočen film s názvem `title`.“

Pokud θ je vždy splněná podmínka 1, pak $\pi_{S_2}(\sigma_1(C_{collection})) = \pi_{S_2}(C_{collection})$. Proto:

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

```
[
  { title: 'Gladiator' },
  { title: 'Gladiator' },
  { title: 'X-Men' },
  { title: 'The Matrix' }
]
```

je kolekce s charakteristickou vlastností „`title` je název filmu.“

4 Mazání a změna dokumentů

Vezměme kolekční proměnnou `collection` a podmínku θ nad S . Pak příkaz

```
db.collection.deleteMany( $\theta$ )
```

nastaví hodnotu proměnné *collection* na $\sigma_{-\theta}(C)$, kde C je původní hodnota proměnné *collection*. Například:

```
> db.movie.find()

[
  { _id: 1, title: 'Gladiator', year: 1992 },
  { _id: 2, title: 'Gladiator', year: 2000 },
  { _id: 3, title: 'X-Men', year: 2000 },
  { _id: 4, title: 'The Matrix', year: 1999 }
]
```

```
> db.movie.deleteMany({ title: "Gladiator" })
> db.movie.find()
```

```
[
  { _id: 3, title: 'X-Men', year: 2000 },
  { _id: 4, title: 'The Matrix', year: 1999 }
]
```

Součin. Vezměme schémata S_1 a S_2 taková, že $Y(S_1)$ a $Y(S_2)$ jsou disjunktní. **Součinem kolekcí** C_1 nad S_1 a C_2 nad S_2 myslíme kolekci

$$C_1 \times C_2 = \{t_1 t_2 \mid t_1 \in C_1 \text{ a } t_2 \in C_2\}.$$

Pokud $V_1(t_1)$ je charakteristická vlastnost kolekce C_1 a $V_2(t_2)$ kolekce C_2 , pak $C_1 \times C_2$ má charakteristickou vlastnost $V(t)$ rovnou „ $V_1(t)$ a $V_2(t)$.“

Například součinem

```
[ { title: 'Gladiator'}, { title: 'X-Men' } ]
```

a

```
[ { year: 1992 }, { year: 2000 } ]
```

dostaneme:

```
[
  { title: 'Gladiator', year: 1992 },
  { title: 'Gladiator', year: 2000 },
  { title: 'X-Men', year: 1992 },
  { title: 'X-Men', year: 2000 }
]
```

Pokud by první kolekce byla dána: „Řetězec `title` je název filmu.“ a druhá: „V roce `year` byl vydán film.“, pak je třetí kolekce dána: „Řetězec `title` je název filmu a v roce `year` byl vydán film.“

Změna dokumentů. Vezměme kolekční proměnnou `collection`, podmínku θ nad S a dokument t nad $S' \subseteq S$, kde $_id \notin Y(S')$. Pak příkaz:

```
db.collection.updateMany( $\theta$ , { $set:  $t$  })
```

nastaví hodnotu proměnné `collection` na

$$(\pi_{S-S'}(\sigma_{\theta}(C)) \times \{t\}) \cup \sigma_{\neg\theta}(C)$$

kde C je původní hodnota proměnné `collection`. Například:

```
> db.movie.find()

[
  { _id: 3, title: 'X-Men', year: 2000 },
  { _id: 4, title: 'The Matrix', year: 1999 }
]

> db.movie.updateMany({
  title: "The Matrix"
}, {
  $set: { title: "Fight Club" }
})

> db.movie.find()

[
  { _id: 3, title: 'X-Men', year: 2000 },
  { _id: 4, title: 'Fight Club', year: 1999 }
]
```

Otázky a úkoly na cvičení

1. Určete výrokové formy následujících podmínek a запиšte je ve formátu JSON. Tam, kde je to možné, použijte v zápisu podmínek zkratky.
 - (a) `name = "Stanley Kubrick"`
 - (b) `age \neq 70`
 - (c) `born = 1928 \wedge nation = "USA"`
 - (d) `born = 1928 \wedge nation \neq "USA"`

- (e) $\text{born} = 1928 \vee \text{born} = 1920$
 (f) $\neg(\text{born} = 1928 \vee \text{born} = 1920)$
2. Určete podmínky, které mají následující výrokové formy a zapište je ve formátu JSON.
- (a) „name je David Lynch“
 (b) „name není David Lynch“
 (c) „name je David Lynch a born není 1945“
 (d) „name je David Lynch nebo Stanley Kubrick“
 (e) „není pravda, že nation je USA“
 (f) „jestliže name je David Lynch, pak born je 1946“
3. Uvažujme proměnnou `director` s charakteristickou vlastností „`_id` je režisér, který se jmenuje `name` a byl narozen v zemi `nation` roku `born`.“ s hodnotou:

```
[
  { _id: 0, name: 'David Lynch', nation: 'USA', born: 1946 },
  { _id: 1, name: 'Stanley Kubrick', nation: 'USA', born: 1928 },
  { _id: 2, name: 'Federico Fellini', nation: 'Italy', born: 1920 },
  { _id: 3, name: 'Akira Kurosawa', nation: 'Japan', born: 1910 },
]
```

Zapište kolekční výrazy počítající restriktce hodnoty proměnné `director` podle podmínek z prvního a druhého úkolu. Určete hodnoty a charakteristické vlastnosti výrazů. Ověřte hodnoty kolekčních výrazů v MongoDB. Následující úkoly používají kolekční proměnnou `director`.

4. Napište kolekční výrazy, které mají následující charakteristické vlastnosti.
- (a) „name je jméno režiséra, který se narodil v roce 1946.“
 (b) „Ze země `nation` pochází nějaký režisér.“
 (c) „`born` je rok narození Davida Lynche nebo Akira Kurosawy.“
 (d) „V roce `year` se narodil režisér, který není Federico Fellini.“
 (e) „Režisér `_id` není z USA ani z Japonska.“

Určete hodnoty kolekčních výrazů a ověřte je v MongoDB.

5. Napište charakteristickou vlastnost následujícího kolekčního výrazu a určete její hodnotu. Hodnotu ověřte v MongoDB.

```
db.director.find({
  $or: [{
    nation: { $neq: "USA" }
  }, {
    born: 1928
  }]
}, { _id: 0, name: 1})
```

6. Smažte všechny režiséry narozené v USA. Následně smazané režiséry vraťte do databáze.
7. Změňte rok narození Akira Kurosawy na chybný údaj 1920. Chybu v zápětí opravte.