

Pokročilá analýza dat pomocí jazyka DAX

1

Princip práce s daty

- Při vytváření datového modelu je vhodné postupovat v rámci těchto priorit:
 - Vytvoření aktivní relace 1:1 či M:1.
 - Vytvoření aktivní relace M:N
 - Vytvoření neaktivní relace
 - Nepropojení tabulek

Pozn. Nepropojení tabulek výrazně navyšuje požadavky na výpočetní výkon.
- Před vytvářením samotného reportu je žádoucí upravit datový model tak, aby splňoval základní požadavky na datovou integritu (primární klíče, cizí klíče). V ideálním případě by datový model měl mít podobu hvězdicového schématu.
- Odměnou Vám bude výrazně jednodušší náročnost výpočtů v samotném DAX a snazší orientace v rámci dat s přihlédnutím k principu rozšířených tabulek.

2

Pokročilá analýza dat pomocí jazyka DAX

ictPRO

Rozšířená tabulka

data

STATION	NAME	DATE	PRCP	SNWD	TAVG	TMAX	TMIN
EZM00011723	TURANY	09.02.2000	0		5,2		
EZM00011723	TURANY	19.04.2002	0		11,7		
EZM00011723	TURANY	13.07.2002	0		21,1		
EZM00011723	TURANY	22.09.2002	0		11,3		
EZM00011723	TURANY	01.10.2002	0		11		
EZM00011723	TURANY	25.10.2002	0		6,9		
EZM00011723	TURANY	17.11.2002	0		11,5		
EZM00011723	TURANY	28.11.2002	0		7,7		
EZM00011723	TURANY	05.12.2002	0		0,9		

dim datumy

Date	Rok	Ctvrletí	Mesic	Zacatek mesice	Den
09.02.2000	2000	1	2	01.02.2000	9
19.04.2002	2002	2	4	01.04.2002	19
13.07.2002	2002	3	7	01.07.2002	13
22.09.2002	2002	3	9	01.09.2002	22
01.10.2002	2002	4	10	01.10.2002	1
25.10.2002	2002	4	10	01.10.2002	25
17.11.2002	2002	4	11	01.11.2002	17
28.11.2002	2002	4	11	01.11.2002	28
05.12.2002	2002	4	12	01.12.2002	5

dim lokality

Lokalita	Mesto	Zeme
TURANY	Brno	Cesko
TURANY	Brno	Cesko
TURANY	Brno	Cesko
TURANY	Brno	Cesko
TURANY	Brno	Cesko
TURANY	Brno	Cesko
TURANY	Brno	Cesko
TURANY	Brno	Cesko
TURANY	Brno	Cesko

* → 1

* → 1

www.ictpro.cz

3

3

Pokročilá analýza dat pomocí jazyka DAX

ictPRO

Rozšířená tabulka

Datový model bez relací

data

dim datumy

dim lokality

STATION

NAME

DATE

PRCP

SNWD

TAVG

TMAX

TMIN

Date

Rok

Ctvrletí

Mesic

Zacatek mesice

Den

Lokalita

Mesto

Zeme

Datový model s relacemi

data

dim datumy

dim lokality

STATION

NAME

DATE

PRCP

SNWD

TAVG

TMAX

TMIN

Date

Rok

Ctvrletí

Mesic

Zacatek mesice

Den

Lokalita

Mesto

Zeme

vlastní sloupce

sloupce rozšířené tabulky

www.ictpro.cz

4

4

2

Pokročilá analýza dat pomocí jazyka DAX

ictPRO

Rozšířená tabulka

STATION
NAME
DATE
PRCP
SNWD
TAVG
TMAX
TMIN
Date
Rok
Ctvrťletí
Měsíc
Začátek měsíce
Den
Lokalita
Město
Zeme

data

dim datumy

dim lokality

Filtry propadají přes relace do tabulky data

vlastní sloupce

sloupce rozšířené tabulky

www.ictpro.cz

5

5

Pokročilá analýza dat pomocí jazyka DAX

ictPRO

Rozšířená tabulka

STATION
NAME
DATE
PRCP
SNWD
TAVG
TMAX
TMIN
Date
Rok
Ctvrťletí
Měsíc
Začátek měsíce
Den
Lokalita
Město
Zeme

data

dim datumy

dim lokality

SUMX(
ALL(data),
data[PRCP])

Pokud je argumentem funkce ALL() celá tabulka, rušíme filtry v celé rozšířené tabulce

vlastní sloupce

sloupce rozšířené tabulky

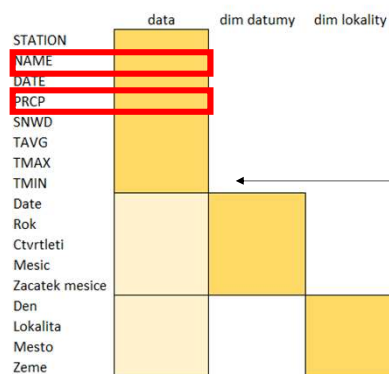
www.ictpro.cz

6

6

3

Rozšířená tabulka



```
SUMX(
    ALL(data[NAME], data[PRCP]),
    data[PRCP])
```

Pokud je argumentem funkce ALL() sloupec či více sloupců, rušíme filtry pouze nad těmito sloupci.

■ vlastní sloupce
■ sloupce rozšířené tabulky

www.ictpro.cz

7

7

Funkce CALCULATE()

- Návrátovou hodnotou je výsledek prvního výrazu (jediný povinný argument funkce)
- Druhým a dalším argumentem jsou filtry upravující data vstupující do výpočtu, tyto filtry mohou být
 - Výrazy logických filtrů
 - Výrazy filtrů tabulek
 - Funkce pro úpravu filtru
- Filtr tabulkových výrazů použije objekt tabulky jako filtr. Může to být odkaz na tabulku modelu, ale pravděpodobněji to bude funkce, která vrací objekt tabulky. Pomocí funkce FILTER můžete zavádět složité podmínky filtrů, včetně těch, které se nedají definovat výrazem logických filtrů.
- Filtr lze upravit i modifikátory, mezi které patří funkce: REMOVEFILTERS(), ALL(), ALLEXCEPT(), ALLNONBLANKROW(), KEEPFILTERS(), USERELATIONSHIP(), CROSSFILTER()

www.ictpro.cz

8

8

Pokročilá analýza dat pomocí jazyka DAX

ictPRO

Příklad využití funkce ALL()

Tabulka data je filtrována přes relace tabulkou 'dim datumy' a 'dim lokality', přičemž v rámci výpočtu se chceme zbavit pouze filtru nad tabulkou 'dim datumy' a následně spočítat maximální hodnotu ve sloupci data[TMAX]

Nefunkční řešení, ve funkci ALL() nelze kombinovat data z více tabulek. Neumíme použít ALL() nad jednou tabulkou a sumu nad druhou tabulkou.

Funkční řešení, zbavujeme se filtru nad rozšířenou tabulkou, filtry z tabulky 'dim datumy' nastavujeme znovu za pomoci funkce FILTER(). Jedná se o zbytečně dlouhý zápis a výpočetně náročný výkon.

Za pomoci funkce CALCULATE() se umíme zbavit filtru nad sloupcem či tabulkou a následně počítat hodnotu ze sloupce jiné tabulky.

```
MAXX(
  ALL(dim_lokalita),
  data[TMAX]
)
```

```
MAXX(
  FILTER(
    ALL(data),
    data[DATE] IN VALUES(data[DATE])),
    data[TMAX]
  )
)
```

```
CALCULATE(
  MAX(data[PRCP]),
  ALL(dim_lokalita))
)
```

www.ictpro.cz

9

9

Pokročilá analýza dat pomocí jazyka DAX

ictPRO

Funkce ALL()

Funkce	Popis
ALL()	Odebere všude všechny filtry. Funkci ALL() jde použít jen k vymazání filtrů, ale nikoli k vrácení tabulky.
ALL(Tabulka)	Odebere všechny filtry z určené tabulky. V důsledku toho funkce ALL(Tabulka) vrátí všechny hodnoty v tabulce a odebere všechny filtry z kontextu, které by se jinak mohly použít. Tato funkce je užitečná, pokud pracujete s mnoha úrovněmi seskupení a chcete vytvořit výpočet, který vytvoří poměr agregované hodnoty k celkové hodnotě. První příklad ukazuje tento scénář.
ALL (Sloupec[; Sloupec[; ...]])	Odebere všechny filtry z určených sloupců v tabulce; všechny ostatní filtry u ostatních sloupců v tabulce zůstanou stále v platnosti. Všechny argumenty sloupců musí pocházet ze stejné tabulky. Varianta ALL(Sloupec) je užitečná, když chcete odebrat kontextové filtry pro jeden nebo více konkrétních sloupců a všechny ostatní kontextové filtry zachovat. Druhý a třetí příklad ukazují tento scénář.
ALLEXCEPT(Tabulka; Sloupec1 [;Sloupec2]...)	Odebere všechny kontextové filtry v tabulce s výjimkou filtrů, které jsou použity u zadaných sloupců. Toto je praktická funkce pro situace, kdy chcete odebrat filtry z mnoha, ale ne všech sloupců v tabulce.

www.ictpro.cz

10

10

5

Pokročilá analýza dat pomocí jazyka DAX

ictPRO

Příklad využití funkce ALL()

Kalkulovaná míra je filtrována hodnotami ve sloupcích, 'dim lokality'[Lokalita], 'dim datumy'[Mesic], nad 'dim datumy'[Rok]. Počítáme maximální hodnotu ve sloupci data[TMAX], přičemž hledáme maximum pro daný 'dim datumy'[Mesic].

Filtr nad 'dim lokality'[Lokalita]

Filtr nad 'dim datumy'[Mesic]

Filtr nad 'dim datumy'[Rok]

Rok

2000

2001

2002

2003

2004

2005

2006

2007

2008

2009

2010

2011

2012

2013

2014

2015

2016

2017

2018

2019

2020

Lokalita

GRAZ

KOSICE

MOSNOV

PIDING

PRAHA KLEMENTINUM

PRAHA LIBUS

RUZYNE

SALZBURG

TURANY

WIEN

Mesic: Max TMAX vsechny roky a lokality

1

21.40

2

23.00

3

24.10

4

30.50

5

34.10

6

36.60

7

38.50

8

39.60

9

34.10

10

27.80

11

24.10

12

20.20

```
CALCULATE(
  MAX(data[TMAX]),
  ALL(dim_lokalita[Lokalita]),
  ALL('dim datumy'[Rok])
)
```

www.ictpro.cz

11

11

Pokročilá analýza dat pomocí jazyka DAX

ictPRO

Příklad využití funkce ALL()

Zrušený filtr

Zachovaný filtr

Zrušený filtr

STATION

NAME

DATE

PRCP

SNWD

TAVG

TMAX

TMIN

Date

Rok

Ctvrletí

Mesic

Zacatek mesice

Den

Lokalita

Mesto

Zeme

data

dim datumy

dim lokality

vlastní sloupce

sloupce rozšířené tabulky

```
CALCULATE(
  MAX(data[TMAX]),
  ALL(dim_lokalita[Lokalita]),
  ALL('dim datumy'[Rok])
)
```

www.ictpro.cz

12

12

Pokročilá analýza dat pomocí jazyka DAX

ictPRO

Příklad využití funkce ALL()

Graf je filtrován přes relace sloupce 'dim datumy'[Rok], 'dim datumy'[Mesic] a 'dim lokality'[Lokalita]. Chceme spočítat maximální data[TMAX] pro daný 'dim lokality'[Lokalita] a 'dim datumy'[Mesic] pro poslední rok vybrané 'dim lokality'[Lokalita].

Rok

☐ 2000

☐ 2001

☐ 2002

☐ 2003

☐ 2004

☐ 2005

☐ 2006

☐ 2007

☐ 2008

☐ 2009

☐ 2010

☐ 2011

☐ 2012

☐ 2013

☐ 2014

☐ 2015

☐ 2016

☐ 2017

☒ 2018

☐ 2019

☐ 2020

Lokalita

☐ GRAZ

☐ KOSICE

☒ MOSNOV

☐ PIDING

☐ PRAHA KLEMENTINUM

☐ PRAHA LIBUS

☐ RUZYNE

☐ SALZBURG

☐ TURANY

☐ WIEN

Mesic

1

2

3

4

5

6

7

8

9

10

11

12

Max TMAX všechny roky a lokality

21.40

23.00

24.10

30.50

34.10

36.60

36.50

39.60

34.10

27.80

24.10

20.20

Maximální teplota v daném roce vs. maximální teplota v posledním roce

Max TMAX

Max TMAX poslední rok

```
CALCULATE(
    MAX(data[TMAX]),
    FILTER(
        ALL('dim datumy'[Rok]),
        'dim datumy'[Rok] =
        CALCULATE(YEAR(MAX(data[DATE])),
            ALL('dim datumy'))
    )
)
```

Výpočet roku posledního dne pro vybranou 'dim lokality'[Lokalita]

www.ictpro.cz

13

13

Pokročilá analýza dat pomocí jazyka DAX

ictPRO

Příklad využití funkce ALL()

Graf je filtrován přes relace sloupce 'dim datumy'[Date], 'dim datumy'[Rok], 'dim lokality'[Lokalita]. Chceme spočítat průměrnou data[TMAX] pro daný 'dim lokality'[Lokalita] a 'dim datumy'[Mesic], 'dim datumy'[Rok].

Rok

☐ 2000

☐ 2001

☐ 2002

☐ 2003

☐ 2004

☐ 2005

☐ 2006

☐ 2007

☐ 2008

☐ 2009

☐ 2010

☐ 2011

☐ 2012

☐ 2013

☐ 2014

☐ 2015

☒ 2016

☐ 2017

☐ 2018

☐ 2019

☐ 2020

Lokalita

☐ GRAZ

☐ KOSICE

☒ MOSNOV

☐ PIDING

☐ PRAHA KLEMENTINUM

☐ PRAHA LIBUS

☐ RUZYNE

☐ SALZBURG

☐ TURANY

☐ WIEN

Mesic

1

2

3

4

5

6

7

8

9

10

11

12

Max TMAX všechny roky a lokality

21.40

23.00

24.10

30.50

34.10

36.60

36.50

39.60

34.10

27.80

24.10

20.20

Maximální teplota v daném roce vs. maximální teplota v posledním roce

Max TMAX

10 průměrná TMAX mesic

```
CALCULATE(
    AVERAGE(data[TMAX]),
    FILTER(
        ALL('dim datumy'[Date], 'dim datumy'[Mesic], 'dim datumy'[Rok]),
        'dim datumy'[Mesic] = SELECTEDVALUE('dim datumy'[Mesic])
        && 'dim datumy'[Rok] = SELECTEDVALUE('dim datumy'[Rok]))
    )
)
```

Abychom mohli filtr nastavit, je třeba sloupce uvést ve funkci ALL() tak, abychom dostali tabulku obsahující i tyto sloupce.

www.ictpro.cz

14

14

7

Pokročilá analýza dat pomocí jazyka DAX

ictPRO

Informační funkce

Funkce	Popis
ISFILTERED()	Vrátí hodnotu TRUE, pokud je zadaná tabulka nebo sloupec přímo filtrována.
ISCROSSFILTERED()	Vrátí hodnotu TRUE, pokud je zadaná tabulka nebo sloupec křížové filtrování.
HASONEVALUE()	Vrátí hodnotu TRUE, když je kontext pro sloupec columnName vyfiltrovaný jenom na jednu jedinečnou hodnotu. V opačném případě je FALSE.
HASONEFILTER()	Vrátí hodnotu TRUE, pokud je počet přímo filtrovaných hodnot ve sloupci columnName jeden. v opačném případě vrátí hodnotu FALSE. Tato funkce je podobná funkci HASONEVALUE(). Rozdíl je v tom, že funkce HASONEVALUE() pracuje s křížovými filtry, zatímco funkce HASONEFILTER() pracuje s přímým filtrem.

www.ictpro.cz

15

Pokročilá analýza dat pomocí jazyka DAX

ictPRO

Funkce pro práce s tabulkami I

Funkce	Popis
VALUES()	Pokud je vstupním parametrem název sloupce, vrátí tabulku s jedním sloupcem, která obsahuje jedinečné hodnoty ze zadaného sloupce. Duplicitní hodnoty se odeberou a vrátí se jen jedinečné hodnoty. Je možné přidat hodnotu BLANK. Pokud je vstupním parametrem název tabulky, vrátí se tabulka se stejnými sloupci.
DISTINCT ()	Vrátí tabulku s jedním sloupcem, která obsahuje jedinečné hodnoty ze zadaného sloupce. Jinými slovy, duplicitní hodnoty se odeberou a vrátí se jen jedinečné hodnoty.
UNION()	Vytvoří sjednocenou (spojenou) tabulku z dvojice tabulek.
ROW()	Vrátí tabulku s jedním řádkem obsahujícím hodnoty, které jsou výsledkem výrazů zadaných v jednotlivých sloupcích.

www.ictpro.cz

16

Pokročilá analýza dat pomocí jazyka DAX

ictPRO

Funkce pro práci s tabulkami II

Funkce	Popis
GENERATE()	Vrátí tabulku s kartézským součinem každého řádku tabulky table1 s tabulkou, která je výsledkem vyhodnocení tabulky table2 v kontextu aktuálního řádku z tabulky table1.
GENERATEALL ()	Vrátí tabulku s kartézským součinem každého řádku tabulky table1 s tabulkou, která je výsledkem vyhodnocení tabulky table2 v kontextu aktuálního řádku z tabulky table1.
SUMMARIZE()	Vrátí souhrnnou tabulku pro požadované součty sady skupin.
ADDCOLUMNS()	Přidá do dané tabulky nebo výrazu tabulky počítané sloupce.

www.ictpro.cz

17

Pokročilá analýza dat pomocí jazyka DAX

ictPRO

Příklad využití funkce s tabulkami

Potřebujeme rozlišit data pro lokality v Praze (ty si definujeme ručně) a ostatní lokality. Následně k těmto lokalitám doplníme roky ze sloupce 'dim datумы[Rok]. V konečné fázi dopočítáme hodnotu z 'data[TMAX], kdy při výpočtu potřebujeme rozlišit, jestli počítáme hodnotu pro lokalitu, která je v 'data[NAME] nebo ne.

```
UNION(
  ROW("Lokalita", "RUZYNE"),
  ROW("Lokalita", "PRAHA LIBUS"),
  ROW("Lokalita", "PRAHA KLEMENTINUM"),
  ROW("Lokalita", "Ostatni"))
```

Lokalita
RUZYNE
PRAHA LIBUS
PRAHA KLEMENTINUM
Ostatni

```
GENERATE(
  UNION(
    ROW("Lokalita", "RUZYNE"),
    ROW("Lokalita", "PRAHA LIBUS"),
    ROW("Lokalita", "PRAHA KLEMENTINUM"),
    ROW("Lokalita", "Ostatni")),
  VALUES('dim datумы'[Rok]))
```

Lokalita	Rok
RUZYNE	2000
PRAHA LIBUS	2000
PRAHA KLEMENTINUM	2000
Ostatni	2000
RUZYNE	2001
PRAHA LIBUS	2001
PRAHA KLEMENTINUM	2001
Ostatni	2001

```
ADDCOLUMNS(
  GENERATE(
    UNION(
      ROW("Lokalita", "RUZYNE"),
      ROW("Lokalita", "PRAHA LIBUS"),
      ROW("Lokalita", "PRAHA KLEMENTINUM"),
      ROW("Lokalita", "Ostatni")),
    VALUES('dim datумы'[Rok])),
  "Maximalni teplota",
  IF([Lokalita] IN VALUES(data[NAME]),
    CALCULATE(MAX(data[TMAX]), FILTER(data, data[NAME] = EARLIER([Lokalita]) &&
    YEAR(data[DATE]) = EARLIER('dim datумы'[Rok]))),
    CALCULATE(MAX(data[TMAX]), FILTER(data, NOT data[NAME] = EARLIER([Lokalita])
    && YEAR(data[DATE]) = EARLIER('dim datумы'[Rok])))))
```

Lokalita	Rok	Maximalni teplota
RUZYNE	2000	33,6
PRAHA LIBUS	2000	34,8
PRAHA KLEMENTINUM	2000	35,6
Ostatni	2000	37
RUZYNE	2001	31,3

www.ictpro.cz

18

Pokročilá analýza dat pomocí jazyka DAX

ictPRO

Příklad využití funkce s tabulkami II

Máme graf zobrazující maximální 'data[TMAX] v jednotlivých 'dim datumy'[Rok] a 'dim lokality[Lokalita]. Potřebujeme barevně odlišit hodnoty rovnající se maximum z maximálních 'data[TMAX] a minimu z maximálních 'data[TMAX].

Dopočítání nového sloupce, kdy za pomoci funkce CALCULATE() využíváme toho, že tabulky, ze kterých jsme vytvořili sloupce, byly propojeny s tabulkou data.

```
VAR max_teplota =
MAXX(
  ADDCOLUMNS(
    GENERATE(
      ALL('dim datumy'[Rok]),
      DISTINCT('dim lokality'[Lokalita])),
      "Maximalni TMAX", CALCULATE(MAX(data[TMAX])),
      [Maximalni TMAX])
  )
)

VAR min_teplota =
MINX(
  ADDCOLUMNS(
    GENERATE(
      ALL('dim datumy'[Rok]),
      DISTINCT('dim lokality'[Lokalita])),
      "Maximalni TMAX", CALCULATE(MAX(data[TMAX])),
      [Maximalni TMAX])
  )
)

RETURN
SWITCH(TRUE(),
  MAX(data[TMAX]) = max_teplota, "brown",
  MAX(data[TMAX]) = min_teplota, "cyan",
  "gold")
```

www.ictpro.cz

19

Pokročilá analýza dat pomocí jazyka DAX

ictPRO

Funkce pro práce relacemi

Funkce	Popis
RELATED()	Vrátí hodnotu v relaci z jiné tabulky. Funkce RELATED vyžaduje, aby mezi aktuální tabulkou a tabulkou se souvisejícími informacemi existovala relace. Zadáte sloupec, který obsahuje požadovaná data, a tato funkce sledováním existující relace N:1 načte hodnotu ze zadaného sloupce v související tabulce. Pokud relace neexistuje, musíte ji vytvořit.
RELATEDTABLE()	Vyhodnotí výraz tabulky v kontextu upraveném danými filtry. Funkce RELATEDTABLE změní kontext, ve kterém jsou data filtrována, a vyhodnotí výraz v novém kontextu, který zadáte.
USERELATIONSHIP()	Určuje relaci, která se má použít v určitém výpočtu, jako relaci, která existuje mezi sloupcem columnName1 a columnName2.
CROSSFILTER()	Určuje směr křížového filtrování, který se má použít při výpočtu relace, která existuje mezi dvěma sloupci.
TREATAS()	Použije výsledek výrazu tabulky jako filtry na sloupce z nesouvisející tabulky. Pro použití je nejlepší, když mezi tabulkami neexistuje relace. Pokud máte více relací mezi tabulkami, zvažte místo toho použití funkce USERELATIONSHIP.

www.ictpro.cz

20