

**Použití dynamických
parametrů prostředí
SET_ENV_
v MarushkaDesignu**



GEOVAP

OBSAH

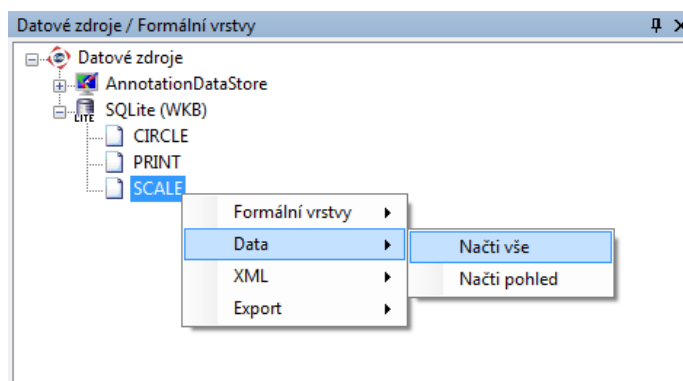
1	CÍL PŘÍKLADU	2
2	PRÁCE S PŘÍKLADEM	2
3	UKÁZKA DIALOGOVÉHO OKNA	3
4	STRUČNÝ POPIS PŘÍKLADU V MARUSHKADESIGNU	5

1 Cíl příkladu

V tomto příkladu si ukážeme práci s dynamickými parametry SET_ENV_ v MarushkaDesignu. Příklad byl vytvořen ve verzi 4.0.1.0 a nemusí být tedy kompatibilní se staršími verzemi.

2 Práce s příkladem

- Do složky **c:\MarushkaExamples** rozbalíme obsah souboru **SET_ENV_CZ.zip**. Cílovou složku je nutné respektovat kvůli provázanosti cest s projektem. V případě umístění souborů do jiné cesty by nebylo možné s příkladem pracovat.
- V prostředí MarushkaDesignu otevřeme projekt **SET_ENV_CZ.xml**.
- Označíme formální vrstvu **SCALE**, v datovém skladu SQLite (WKB), v kontextovém menu vybereme Data – Načti vše:



- V mapovém okně zvolíme zobrazit „Vše“:

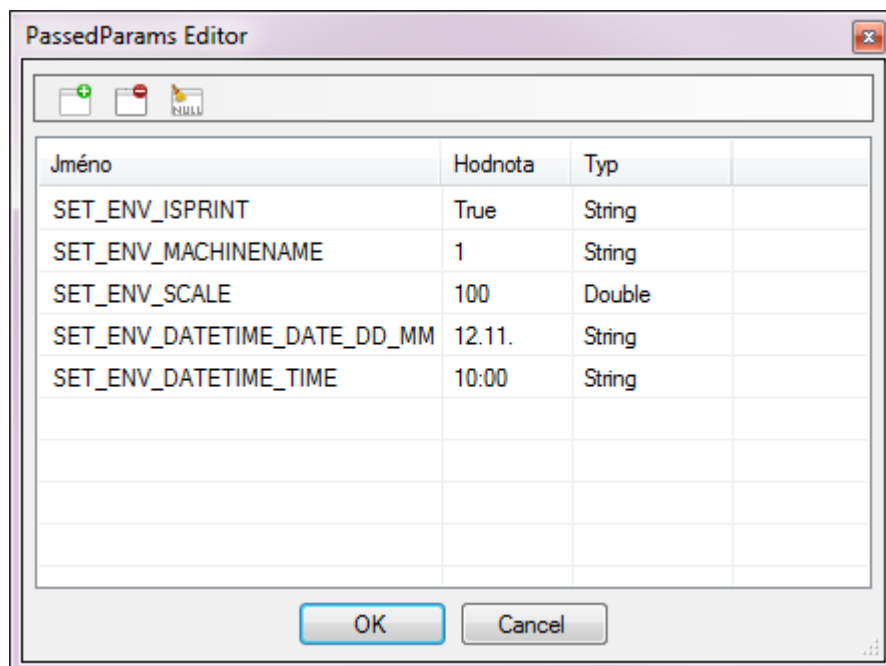


- Spustíme lokální webový server:

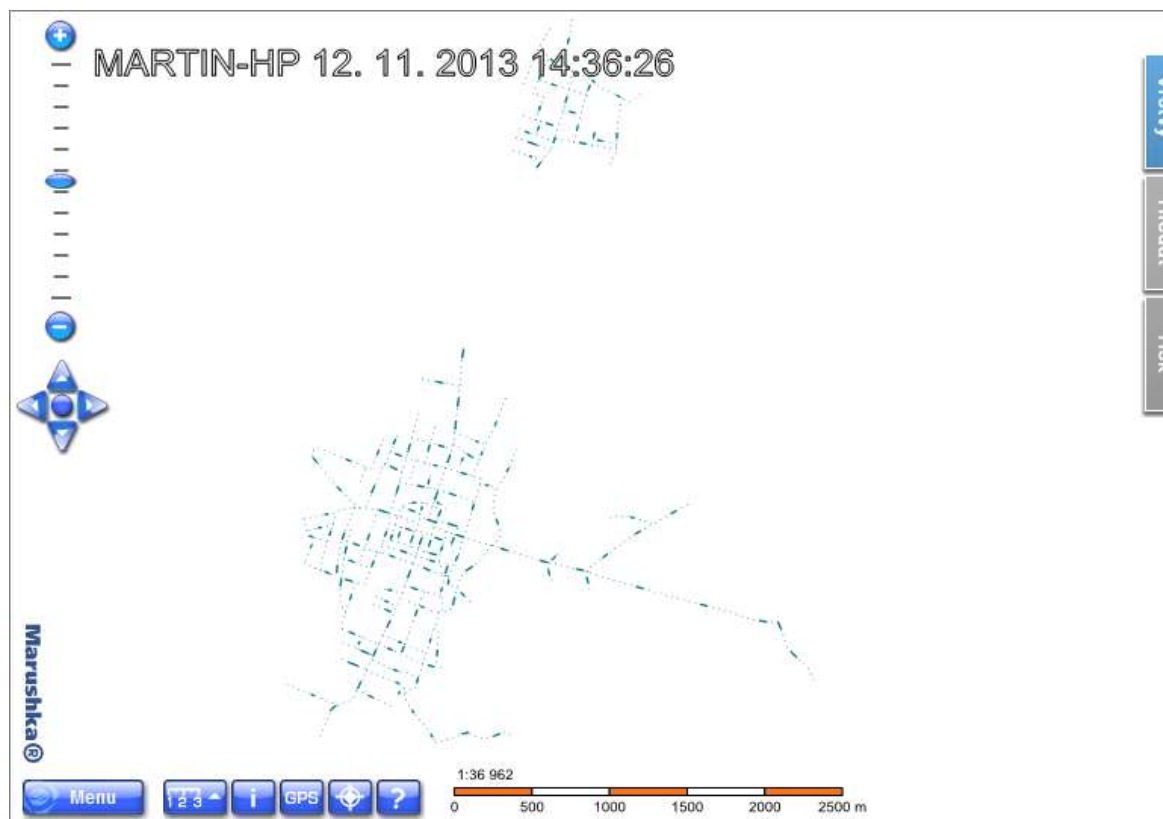


3 Ukázka dialogového okna

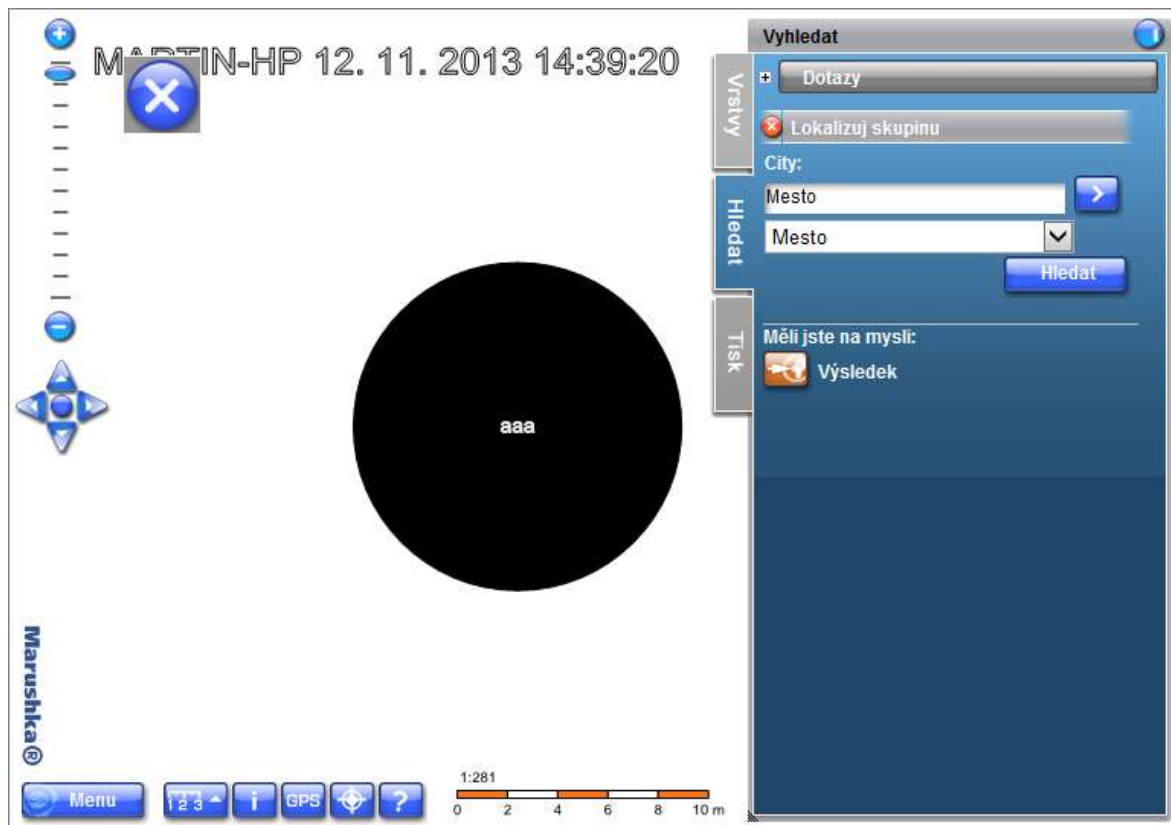
Obr 1: Ukázka definice parametrů SET_ENV_ v PassedParams editoru



Obr 2: Ukázka zobrazení formální vrstvy SCALE v lokálním webovém serveru



Obr 3: Ukázka výsledku lokalizačního dotazu *Lokalizuj skupinu*



4 Stručný popis příkladu v MarushkaDesignu

Dynamické parametry SET_ENV_

Jednotlivé parametry *SET_ENV_* jsou parametry prostředí a slouží k substituci do jednotlivých databázových dotazů. V případě použití některého z těchto parametrů je nutné takový parametr rovněž definovat do *PassedParams* projektu a uvést u něj defaultní hodnotu. Bez této definice by se mohlo stát, že server nebude hodnotu parametru *SET_ENV_* znát a výsledek by tak mohl skončit chybovým stavem.

Hodnoty těchto parametrů není možno měnit uživatelem zvenku, jsou dynamicky nahrazovány serverem (jejich hodnota se v čase mění) v případě, že se vyskytnou v databázovém dotazu (např. vyvolání tiskové úlohy u *SET_ENV_ISPRINT*).

Zde je seznam podporovaných *SET_ENV_* parametrů v aktuální verzi MarushkaDesignu (verze 4.0.1.0):

SET_ENV_SCALE: Parametr sloužící k definici měřítkového rozsahu zobrazované vrstvy, formát: *Double*.

SET_ENV_ISPRINT: Parametr, který umožňuje jiné chování publikace v případě, kdy je vyvolána tisková úloha, formát: *Boolean*.

SET_ENV_MACHINENAME: Parametr obsahující jméno uživatelského počítače, formát: *String*.

SET_ENV_THEMEID: Parametr obsahující ID aktuálního tématu, formát: *Int*.

SET_ENV_USER: Parametr, který dokáže pracovat s uživatelským jménem přihlášeného uživatele (v případě autorizovaného přístupu do databázového datového skladu), formát: *String*.

SET_ENV_DATETIME_DATE_DD_MM: Parametr obsahující aktuální datum ve formátu: *Den-Den Měsíc-Měsíc Rok*, formát: *String*.

SET_ENV_DATETIME_DATE_MM_DD: Parametr obsahující aktuálním datum ve formátu: *Měsíc-Měsíc Den-Den Rok*.

SET_ENV_DATETIME_TIME: Parametr obsahující aktuálním čas ve formátu: *Hodina-Hodina Minuta-Minuta Sekunda-Sekunda*.

Příklad obsahuje celkem 3 formální vrstvy a 1 anotační vrstvu. Konkrétní využití dynamických parametrů *SET_ENV_* je uvedeno v následujících bodech:

1. Parametr **SET_ENV_ISPRINT**. Formální vrstva *PRINT* zobrazuje uliční čáry tří obcí s popisy. V *DBCColumnsToClient* má tato formální vrstva nastaven následující řetězec:

```
~(string)SET_ENV_ISPRINT~,CASE WHEN ~(string)SET_ENV_ISPRINT~='True' THEN '255 255 128 0' ELSE '255 125 125 0' END SET_PARS_RGBCOLOR
```

To znamená, že po vyvolání tiskové úlohy na kartě „*Tisk*“ a kliku na tlačítko „*Další*“ se všechny elementy zobrazované formální vrstvy přebarví na oranžovou barvu. Stejným způsobem by bylo možné nastavit, aby se v tisku zobrazovaly například pouze liniové prvky z databáze nebo naopak, aby se do tisku přidal nějaký nový prvek (např. vodotisk). Tento parametr má jako hlavní typ nastaven *Bool*, nicméně ho lze úspěšně použít i když je typu *String* (takto je použit v příkladu).

Kdyby nebyla defaultní hodnota parametru *SET_ENV_ISPRINT* definovaná v *PassedParams* projektu, nebylo by možné s formální vrstvou *PRINT* provést například operace *Data – Načti vše* nebo *Data – Načti pohled*, protože by server nevěděl, jakou hodnotu má místo parametru dosadit a výsledek by tak skončil syntax errorem.

2. Parametr **SET_ENV_MACHINENAME** je nahrazován jménem uživatelského počítače. Zde je tento parametr zobrazován pomocí anotační vrstvy *Annotation*. Samotná anotace je potom zobrazována v levém horním rohu mapového okna společně s vrstvami *SCALE* a *PRINT*, konkrétně v tomto příkladu zobrazuje text MARTIN-HP. Způsob použití tohoto parametru v anotační vrstvě je poněkud nestandardní, protože se zde nevkládá do položky vlastností *DBCOLUMNSToClient*, ale do položky *Text*, do které byl vložen řetězec `~SET_ENV_MACHINENAME~`. V této verzi MarushkaDesignu je však možné vložit tento parametr do anotační vrstvy pouze bez typu, jinak by nebyl nahrazen názvem uživatelského počítače, ale byl by zobrazen pouze zadaný řetězec.

Můžete si sami zkusit, zda se Vám zobrazuje správné jméno počítače. Ve Windows 7 nebo 8 klikněte na nabídku *Start* / pravé tlačítko na *Počítač* / *Vlastnosti* / a zde je položka *Název počítače*, tento název by měl odpovídat názvu zobrazenému pomocí anotační vrstvy.

3. Parametr **SET_ENV_SCALE** je nahrazován hodnotou aktuálního měřítka mapy. Zde je konkrétně použit u formální vrstvy *SCALE*, která má v *DBCOLUMNSToClient* pomocí *CASE* nastavenou dynamickou změnu barvy zobrazovaných elementů uliční sítě při změně měřítka. Konkrétně je zde nastaveno celkem 5 kategorií, každá kategorie má nastaven jiný rozsah měřítka a jinou barvu zobrazení. Celý SQL dotaz, pomocí kterého je toto chování vrstvy nastaveno, je definován v *DBCOLUMNSToClient* formální vrstvy *SCALE*.

Celý dotaz má tuto podobu:

```
~(double)SET_ENV_SCALE~, CASE WHEN ~(double)SET_ENV_SCALE~<5000 THEN
'255 255 0 0' WHEN ~(double)SET_ENV_SCALE~<10000 THEN '255 0 0 255'
WHEN ~(double)SET_ENV_SCALE~<25000 THEN '255 255 0 255' WHEN
~(double)SET_ENV_SCALE~<50000 THEN '255 0 125 125' ELSE '255 0 255
0' END SET_PARS_RGBCOLOR
```

Sami si můžete vyzkoušet funkčnost změny barev při přibližování, resp. oddalování pomocí kolečka myši při zapnuté vrstvě *SCALE* v lokálním webovém serveru.

4. Opět pomocí parametru **SET_ENV_SCALE** bylo umožněno vytvořit složitější lokalizační dotaz, *Lokalizuj skupinu*. Tento dotaz by bez použití tohoto parametru musel být vytvořen pomocí dvou formálních vrstev.

Tento lokalizační dotaz má nastaveno, že výsledek dotazu lokalizace zobrazuje černý kruh v reálné velikosti a do jeho referenčního bodu vykresluje text 'aaa', který má nastavenou pevnou pixelovou velikost 10 pixelů. Tento text je však vykreslován pouze do měřítka 1 : 2 500, protože při vyšším měřítku by byl text větší než buňka, resp. by byla buňka menší než text.

V tomto dotazu je nejdůležitější konstrukce *UNION ALL*, která spojí obě části dotazu. Nevýřazuje však duplicitu, proto je při jejím použití třeba pamatovat také na to, že spojované dotazy musí vracet stejné sloupce. Pomocí použití pseudosloupců byl nastaven referenční bod pro zobrazování textu, barva výplně, text, výška textu, nastavení výšky textu v pixelech a jméno buňky, která je zobrazována ve výsledku lokalizace.

V případě, že by nebyla uvedena defaultní hodnota pro parametr *SET_ENV_SCALE* v *PassedParams* projektu, tak by tento dotaz nebyl funkční. Před vyhodnocením dotazu, v okamžiku kdy jsou vyhledávány možné výskyty, totiž není dostupné aktuální měřítka, a proto je namísto aktuálního měřítka mapy dosazena defaultní hodnota z *PassedParams*, která je porovnávána s podmínkou ve *WhereClause*. Při vyhodnocení výsledku je již s podmínkou porovnávána aktuální hodnota měřítka.

Celý dotaz má tuto podobu:

```
select xmin-20 XMIN, ymin-20 YMIN, xmax+20 XMAX, ymax+20 YMAX, id
ID, '0 0 7' SET_PARS_POINT_FROM_CORG, '' SET_PARS_RGBFCOLOR, geom
GEOM, null SET_PARS_TEXT, 'Bunka' SET_PARS_CELLNAME, null
SET_PARS_HEIGHT, 'FALSE' SET_PARS_PIXELSIZE FROM GS_TABLE WHERE City
like '~1~'
```

UNION ALL

```
select xmin-20 XMIN, ymin-20 YMIN, xmax+20 XMAX, ymax+20 YMAX, id
ID, '0 0 7' SET_PARS_POINT_FROM_CORG, '255 255 255 255'
SET_PARS_RGBFCOLOR, geom GEOM, 'aaa' SET_PARS_TEXT, null
SET_PARS_CELLNAME, 10 SET_PARS_HEIGHT, 'TRUE' SET_PARS_PIXELSIZE FROM
GS_TABLE WHERE City like '~1~' AND ~SET_ENV_SCALE~<2500
```

5. Od verze Marushky 3.0.16.0 je možné použít tři nové typy parametrů SET_ENV_. Jedná se o SET_ENV_DATETIME_DATE_DD_MM, SET_ENV_DATETIME_DATE_MM_DD a SET_ENV_DATETIME_TIME.

Novinkou je také možnost kombinace parametrů SET_ENV_, kdy je možné libovolně je kombinovat buď v databázových dotazech, například ve vlastnosti *Text* anotační vrstvy. Názorná ukázka je uvedena u anotační vrstvy, která má ve vlastnosti *Text* definován řetězec:

```
~SET_ENV_MACHINENAME~ ~SET_ENV_DATETIME_DATE_DD_MM~
~SET_ENV_DATETIME_TIME~
```

a zobrazuje tedy v levém horním rohu aktuální jméno uživatelského počítače, datum ve formátu: *Den-Den Měsíc-Měsíc* a čas ve formátu: *Hodina-Hodina Minuta-Minuta Sekunda-Sekunda*. Ještě je potřeba mít u anotační vrstvy nastavenou položku *Type* na *Text*, aby se nahrazené parametry zobrazily.