

3D VIZUALIZACE MODELŮ TERÉNU POMOCÍ WEBOVÝCH TECHNOLOGIÍ

Lukáš HERMAN

Geografický ústav, Přírodovědecká fakulta, Masarykova univerzita, Kotlářská 2, 611 37, Brno, Česká republika
herman.lu@mail.muni.cz

Abstrakt

V současné době existuje značné množství technologií umožňujících vizualizaci 3D dat ve webovém prohlížeči. U většiny z nich se však projevují problémy při vykreslování objemných dat, jakými jsou kupříkladu modely terénu. Tento příspěvek se právě z toho hlediska zaměřuje na technologii X3DOM. X3DOM využívá pro uložení prostorových dat strukturu formátu X3D, modely terénu zde lze uložit pomocí elementů IndexedFaceSet a ElevationGrid. V X3DOM však existují i možnosti jak modely terénu uložit úspornějším způsobem, například pomocí prvků BinaryGeometry nebo ImageGeometry. Oba prvky jsou založeny na binárním uložení 3D dat. Dalším možností je element BVHRefiner, což je komponenta, která adaptivně zjemňuje daný model terénu a dynamicky načítá jednotlivé části jeho kvadrantového stromu. Pro optimalizaci modelů terénu byly použity open source nástroje BVHRefiner Dataset Converter a AOPT, součást frameworku InstantReality. V příspěvku jsou způsoby uložení digitálních modelů terénu srovnány jak z hlediska velikosti výsledných dat, tak rychlosti jejich vykreslování. Vzájemně porovnány jsou rovněž dvě různé datové struktury používané v rámci prvku BVHRefiner, tzv. TREE a WMTS (Web Map Tile Service). V závěru jsou shrnuty zjištěné poznatky a naznačeny možné směry dalšího vývoje.

Abstract

Considerable amount of technologies enabling the visualization of 3D data in a web browser is currently available. However problems experience for most of them when rendering large data sets such as terrain models. This paper is focused from this point of view on the technology X3DOM. X3DOM uses for storage of spatial data structure of the X3D format, so terrain models can be saved here through elements IndexedFaceSet and ElevationGrid. There are also different ways how to store digital terrain models more economically in X3DOM, this option utilizes elements such as BinaryGeometry or ImageGeometry. Both elements are based on binary storage of 3D data. Another option is BVHRefiner element, which is a component that adaptively refines the terrain model and dynamically loads the individual parts of the quadtree. Open source tools BVHRefiner Dataset Converter and AOPT, a component of InstantReality framework, were used for optimization of terrain models. Methods of storing digital terrain models are compared in term of data size as well as their rendering speed. Two different data structures used within the element BVHRefiner, so called TREE and WMTS (Web Map Tile Service), are mutually also compared. The conclusion summarizes the findings and suggested possible future directions.

Klíčová slova: 3D vizualizace; digitální model terénu; webová kartografie; X3D.

Keywords: 3D visualization; digital terrain model; web cartography; X3D.

ÚVOD

Významným momentem v procesu 3D vizualizace geografických dat je v první řadě samotné zobrazení virtuálního 3D prostředí a objektů v něm. Více než teoretickými poznatky je oblast 3D vizualizace prostorových dat ovlivněna technologiemi a jejich možnostmi (Wood et al. 2005). Pro 3D vizualizaci prostorových dat přitom vzniká v současné době řada jak desktopových, tak internetových systémů a technologií (Voženílek 2005).

Tento příspěvek popisuje možnosti technologií určených pro vykreslování 3D grafiky přímo ve webovém prohlížeči. Zaměřuje se především na technologii X3DOM a uložení dat o modelech terénu v této JavaScript knihovně. Modely terénu mohou tvořit podklad pro ortofotomapy nebo i jiná data, kupříkladu hlukové mapy

(Herman, Řezník 2013). Důležitým aspektem při 3D vizualizaci modelů terénu je zejména jejich komprese, tak aby byl objem přenášených dat co nejmenší a zároveň byly data dostatečně rychle vykreslována.

3D VIZUALIZACE PROSTOROVÝCH DAT VE WEBOVÉM PROHLÍŽEČI

Stávající webové technologie jako HTML (*HyperText Markup Language*) nebo SVG (*Scalable Vector Graphic*) nativně nepodporují zobrazování 3D objektů. Přesto však došlo ke vzniku několika řešení, která dokáží pomocí 2D elementů vytvořit dojem třetí dimenze. Významnější rozšíření možností vykreslování 3D grafiky přímo v prohlížeči souvisí až se zaváděním standardu HTML5 (Behr et al. 2009). HTML5 je nová verze jazyka HTML, v současnosti (k 20. 12. 2014) ve stádiu *W3C Recommendation*. Je zpracovávána organizacemi W3C (*World Wide Web Consortium*) a WHATWG (*Web Hypertext Application Technology Working Group*). Starší vývojová verze HTML5 specifikace (z 23. dubna 2009) také zmiňuje, jak by měla být do webových stránek vkládána 3D grafika. V kapitole 12.2 specifikace je zde uvedeno, že vkládání 3D grafiky do HTML je záležitostí X3D (*eXtensible 3D*) nebo technologií na X3D založených. Specifikace však neuvádí, jak by integrace 3D dat do HTML měla konkrétně vypadat, zda by tato data měla být definována přímo v HTML struktuře nebo jestli slouží prvky DOM (*Document Object Model*) jen jako datové kontejnery (Behr et al. 2009). V novějších verzích standardu HTML5 již problematika 3D grafiky není vůbec zmiňována (Brown et al. 2014).

Další technologií, která navazuje na HTML5 je WebGL. WebGL vytvořila organizace Khronos Group. Vychází ze standardu OpenGL a programovacího jazyka JavaScript, první prototyp vznikl v roce 2006. Slouží k zobrazování 3D grafiky přímo ve webové stránce. Tato grafická knihovna může fungovat jen v rámci elementu *canvas* (tedy pouze v HTML5). Pro fungování WebGL je nezbytná jeho podpora grafickou kartou a dále je nutný webový prohlížeč, který umí pracovat s HTML5 a podporuje samotné WebGL. Za hlavní nedostatek WebGL jsou uváděny bezpečnostní rizika. WebGL totiž přistupuje až ke grafickým ovladačům a grafické paměti počítače (Lienert et al. 2012).

Z WebGL vychází i další rozšíření a modifikace, jež by se daly přirovnat ke knihovně *jQuery*. Tyto rozšíření zpravidla obsahují sady 2D a 3D grafických tříd, které blíže charakterizující scény či jednotlivé objekty, a vývojář tak nemusí manipulovat přímo 3D matematickými strukturami ve WebGL. Takovým rozšířením je například C3DL (*Canvas 3D JS Library*). Dalšími obdobnými systémy pro integraci 3D grafiky do HTML jsou XML3D, GLGE, PhiloGL nebo SceneJS (Behr et al. 2010).

X3DOM

Hlavním principem knihovny X3DOM je „živá“ X3D datová struktura ukládaná a přístupná ve struktuře HTML DOM. Se scénou se manipuluje pomocí pouhého přidávání, mazání a modifikace uzlů DOM. Přitom nejsou vyžadovány žádné speciální plug-iny a jsou podporovány obvyklé HTML události (např. *onclick*). Systém X3DOM je tvořen třemi základními prvky: *User Agent* (internetový prohlížeč) udržuje DOM strom, integruje finální obraz a poskytuje určitý typ *URI-resolveru* pro stahování rastrových obrazů, videí, zvuku a dalších prvků scény; *X3D runtime* poskytuje funkce pro vyvážení a změny X3D scény, vykresluje tuto scénu při jakémkoliv změně a umožňuje navigaci a uživatelské vstupy; *Connector* tvoří jádro celé architektury, propojuje strom DOM s *X3D runtime*, důležitou funkcí je pak především přenos změn v obou směrech. Implementace systému X3DOM je realizovaná v JavaScriptu. Tato vrstva monitoruje DOM strom i X3D elementy (Behr et al., 2010).

Tab. 1. Základní grafická primitiva podporovaná v X3DOM

3D primitiva	<i>Box, Cone, Cube, Sphere</i>
2D primitiva	<i>Arc, Circle, Disk, Polyline, Polypoint, Rectangle, TriangleSet, Text</i>
Komplexní prvky	<i>IndexedFaceSet, IndexedLineSet, PointSet, ElevationGrid</i>

Pro samotné vykreslení 3D scény je používána jedna ze čtyř *backend technologií*. *Backend technologie* jsou uspořádány do tzv. *fallback modelu*. Ten je uspořádává do série podle rychlosti vykreslování. Cílem *fallback modelu* je vykreslit 3D data pomocí nejrychlejší dostupné technologie. Jednotlivé *backend technologie* se liší

nejen rychlostí vykreslování 3D scén, ale i rozšířením mezi potenciálními uživateli. Zatímco nativní podporu X3D nebo X3D plug-iny používá minimum uživatelů webu, backend technologie, které vykreslují 3D scény pomaleji (WebGL a Flash 11 plug-in) jsou daleko rozšířenější (Behr et al. 2010; Herman 2013).

Jak již bylo zmíněno, knihovna X3DOM využívá vybranou část elementů z formátu X3D. Tyto 3D data mohou být uloženy přímo v HTML kódu nebo je lze uložit do externích X3D souborů a odkazovat na ně prostřednictvím elementu *Inline*.

Jádro X3D představují elementy pro vytvoření geometrických objektů. Tyto elementy lze rozdělit do tří kategorií, na 3D prvky, 2D prvky a komplexní prvky. Konkrétní grafické elementy z jednotlivých kategorií jsou uvedeny v tab. 1. 3D i 2D primitiva lze při vizualizaci prostorových dat použít například pro proporcionální symboly, element *Text* pak pro popis geoprvků. Pro komplexnější geografické objekty, jako jsou například modely terénu, jsou vhodnější elementy *IndexedFaceSet*. Tento prvek je složen ze seznamu souřadnic (atribut *point*) a charakteristiky pořadí, v jakém mají být propojeny (atribut *coordIndex*). Hodnota minus jedna znamená, že je daná geometrie ukončena. Lze tak v rámci jednoho elementu vytvořit množinu nespojitých polygonů.

```
<Shape>
  <IndexedFaceSet coordIndex="0 1 2 3 0 -1" convex="false" solid="false">
    <Coordinate point="-601499.37874879 -1162594.1042506 283.765 ..."/>
  </IndexedFaceSet>
</Shape>
```

Komplexní prvky jsou principiálně založeny na seznamu souřadnic. Tímto se blíží pojetí geometrie obvykle ve vektorovém GIS (data jsou uložena jako body, linie či plochy). Výjimku však představuje *ElevationGrid*, jenž vychází z rasterového přístupu ukládání dat a je používán pro ukládání modelů povrchu či terénu. Souřadnice jsou v X3D ukládány v pořadí X, Y, Z. Kladný směr osy Y směřuje vzhůru.

ÚPRAVY PRO SNÍŽENÍ OBJEMU DAT

Jedním z hlavních nároků kladených na 3D scénu je její rychlé a plynulé zobrazení, což vyžaduje dostatečně vysoký počet *fps* (*frames per second*) při vykreslování scény. Proto je vhodné 3D scénu optimalizovat. Obecně se dá říct, že existuje nepřímá úměrnost mezi počtem polygonů a počtem *fps*, proto je zpravidla snahou autora jejich počet přiměřeně snížit.

Úrovně detailu

Koncept úrovně detailu (angl. *Level of Detail* – LoD) má v rámci X3D podobu seskupujícího elementu (párového tagu), do kterého se modely umístí tak, že na nejvyšší úrovni je model s nejvíce detaily a další verze se řadí sestupně pod něj. Tímto způsobem je vykreslení vzdálených objektů výpočetně mnohem méně náročné. Pokud je jejich přechod zvolen vhodně, uživatel si ho ani nemusí všimnout. Pro použití této metody je potřeba vytvořit několik verzí stejného modelu. Automatizované odvozování nižších LoD může mít formu výběru geometrických elementů (to není složité na implementaci, ale takový postup nelze použít ve všech případech), nebo generalizace geometrie, což ovšem vyžaduje vytvoření netriviálních algoritmů.

```
<LOD center='654.379 272.043 534.379' range='10490 20980'>
  <Shape>
    <!--geometrie v nejpodrobnější úrovni detailu-->
    ...
  </Shape>
  <Shape>
    <!--geometrie v méně podrobné (odvozené) úrovni detailu-->
    ...
  </Shape>
</LOD>
```

Kompresce celých souborů

Velikost X3D dat lze snížit prostřednictvím komprimace dat – uložením celých souborů v binární podobě nebo optimalizací konkrétních elementů, které obsahují dlouhé seznamy souřadnic a tvoří tak zpravidla

největší část datového objemu scény. Na binárně uložené soubory X3D (s příponou *x3db*) lze navíc aplikovat i ZIP kompresi, čímž se docílí dalších snížení velikosti souborů. Binární formu formátu X3D však nelze otevřít ve webové knihovně X3DOM, ale jen v některých desktopových prohlížečích, jako je například *Instant Player*.

Binární uložení X3D prvků

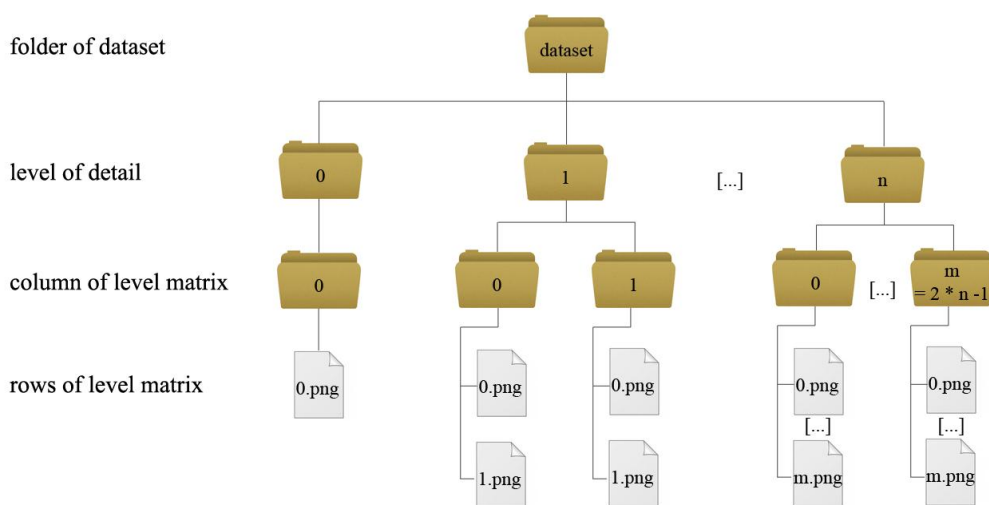
Velikost komplexních prvků (nejčastěji elementu *IndexedFaceSet*) je možné snížit převodem daných geometrických elementů na element *BinaryGeometry*. Ten obsahuje odkazy na seznamy souřadnic (atribut *coord*), normálových vektorů (atribut *normal*) a seznam indexů (atribut *index*). Souřadnice a další číselné hodnoty jsou uloženy v externích binárních souborech s koncovkou *bin*.

Komplexní prvky je také možné zmenšit transformací geometrie do podoby rastrové grafiky (elementu *ImageGeometry*). *ImageGeometry* pracuje s geometrií, jež je uložena v podobě rastrové grafiky. Jsou vytvořeny externí soubory rastrové grafiky, na které je z X3D kódů odkazováno, podobně jako v elementu *BinaryGeometry*. Místo barevných kanálů (R, G, B) jsou v externích rastrových souborech uloženy souřadnice (X, Y, Z).

```
<Shape>
  <BinaryGeometry DEF='BG_6' vertexCount='118 18774' primType='\"TRIANGLESTRIP\"
    \"TRIANGLES\"' position='-603498.5 271.985992432 1163815' size='1309 81.5159912109
    1050' index='bingeo/BG_6_indexBinary.bin' coord='bingeo/BG_6_coordBinary.bin'
    normal='bingeo/BG_6_normalBinary.bin' coordType='Int16' normalType='Int8' />
</Shape>
```

Dynamické načítání 3D dat

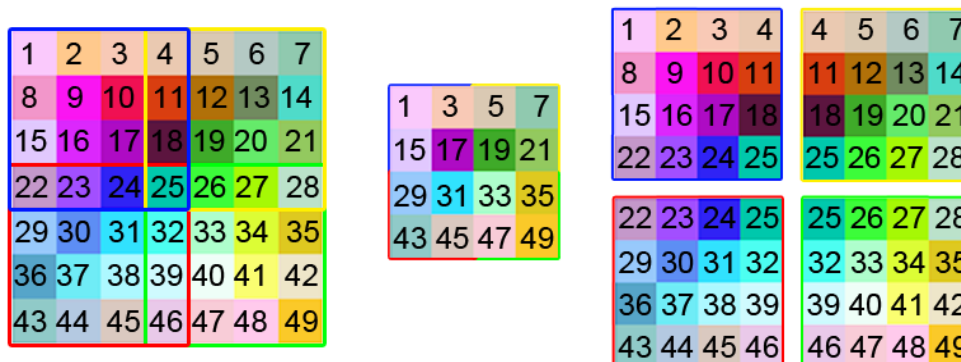
BVHRefiner je prvek, který dynamicky zjemňuje a načítá hierarchická data za běhu. Data mohou být rozčleněna podle dvou různých struktur (*WMTS* a *TREE*). První je založen na OGC specifikaci *WMTS* (*Web Map Tile Service*) a druhá varianta vychází z uspořádání souborů ve složkách. V případě 3D modelů terénu je zatím podporován pouze varianta **WMTS**. Pro každou úroveň detailu je vytvořena vlastní matice dlaždic. Jednotlivé dlaždice jsou uloženy v souborech v dílčích složkách. V případě pěti různých úrovní detailů v aplikaci musí existovat pět složek, číslované od 0 do 4. Na úrovni 0 je jen jediný sloupec i řádek reprezentovaný prostřednictvím složky s názvem 0 (jen jeden obrázek, který reprezentuje celý model terénu). V další úrovni jsou dva sloupce s názvy 0 a 1 (vyjádřené jako podsložky). Na úrovni 1 existují dvě podsložky, každá obsahuje dva obrázky. Na další úroveň obsahují podsložky čtyři obrázky atd. (viz obr. 1).



Obr. 1. Struktura souborů a složek podle schématu WMTS (převzato z: X3DOM 2014)

K vykreslení modelu terénu z dat odpovídající struktury WMTS je nutné specifická konfigurace jednotlivých pixelů v souborech. Sousední dlaždice musí sdílet pixely na hranicích. Obr. 2. ukazuje vlevo původní obrázek, uprostřed obraz na úrovni 0 a na pravé straně čtyři obrázky úrovně 1. Důležitou podmínkou

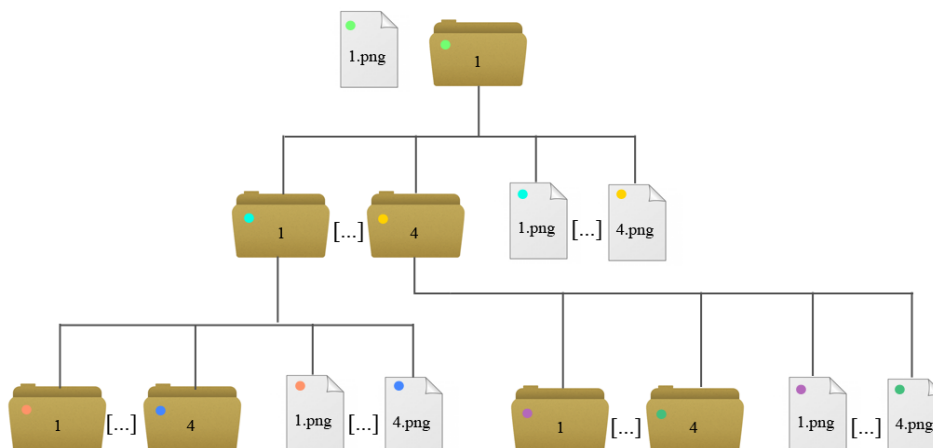
je definován parametr *subdivision*. Kupříkladu když je nastavena hodnota "64 64", má každý dílčí rastrový soubor rozměry 129 krát 129 pixelů. Toto je podmínku algoritmu, aby se zabránilo vzniku trhlin v modelu terénu.



Obr. 2. Uspořádání jednotlivých pixelů v různých úrovních podrobnosti ve struktuře WMTS (převzato z: X3DOM 2014)

Struktura TREE částečně vychází ze stejného schéma jako WMTS. Každá úroveň ve stromu definuje míru podrobnosti modelu terénu. Na úrovni 0 je jeden rastrový obraz (např.: *1.png*), který představuje model terén v nejnižší kvalitě. Dále je zde umístěna složka, která má název 1. V této složce jsou čtyři snímky, které dohromady představují model terén. Rozlišení roste s každou úrovní čtyřnásobně (viz obr. 3). Pozice obrazů pro jemné rozlišení je následující:

- *1.png*: vlevo nahoře
- *2.png*: vlevo dole
- *3.png*: vpravo nahoře
- *4.png*: vpravo dole



Obr. 3. Struktura souborů a složek podle schématu TREE (převzato z: X3DOM 2014)

Tab. 2. Jednotlivé parametry X3DOM elementu *BVHRefiner*

parametr	hodnoty	defaultní	popis
<i>maxDepth</i>	0, 1, ... n	3	maximální hloubka stromu nebo souboru dlaždic
<i>minDepth</i>	0, 1, ... n	0	minimální hloubka stromu, která by měla být renderována nejdříve
<i>interactionDepth</i>	0, 1, ... n	maxDepth	maximální hloubka vykreslovaná během ovládní (procházení) scény
<i>subdivision</i>	0, 1, ... 125	1 1	rozlišení vykreslovaných dlaždic
<i>size</i>	0, 1 ... n	1 1	velikost modelu terénu
<i>factor</i>	0, 1, ... n	1.0	faktor, který ovlivňuje vzdálenost, ve které dochází k vytvoření nebo vykreslení další úrovně (čím vyšší je faktor tím rychlejší vykreslení, čím nižší tím je vyšší kvalita)
<i>maxElevation</i>	0.0, 0.1, ... n	1.0	maximální posun ve směru osy Y
<i>elevationUrl</i>	string	""	URL adresa souboru s modelem terénu
<i>textureUrl</i>	string	""	URL adresa souboru s texturou
<i>normalUrl</i>	string	""	URL adresa souboru s normálovými vektory
<i>elevationFormat</i>	png, jpg, gif ...	png	souborový formát modelu terénu
<i>textureFormat</i>	png, jpg, gif ...	png	souborový formát textury
<i>normalFormat</i>	png, jpg, gif ...	png	formát souboru normálových vektorů
<i>mode</i>	2D, 3D, bin, bvh	3D	typ použitých dat; 2D (rovina), 3D (nahrazení souřadnic Y hodnotou výšky), bin (binární soubory, WMTS), bvh (binární soubory, TREE)
<i>submode</i>	WMTS, TREE	WMTS	použitý systém tvorby dlaždic (WMTS, TREE - zatím jen pro 2D)

TVORBA MODELŮ TERÉNU

Podkladovými daty pro tvorbu digitálních modelů terénu zde byla databáze ZABAGED poskytnutá ČÚZK (Český úřad zeměměřický a katastrální) ve formátu Shapefile. Při práci byla použita výškopisná data (geodetické body, vrstevnice, srázy, mosty, stupně a paty terénních útvarů). Jejich výšková přesnost závisí na sklonu reliéfu v daném místě (pohybuje se 1,5 do 6 m). Data ZABAGED mají souřadnicový systém S-JTSK a výškový systém BpV.

První část zpracování dat probíhala zejména v programu *ArcGIS* verze 10.0, konkrétně především v modulu *ArcScene*. Tento nástroj umožňuje export do formátu VRML (*Virtual Reality Modeling Language*), který lze do X3D dále transformovat. Protože bylo testováno několik variant modelu terénu, byl ze vstupních souborů Shapefile vytvořen nejprve TIN model terénu. K vytvoření posloužila funkce *Create TIN*. Při tvorbě byla dodržována Delaunayho triangulace, vstupními soubory byly vrstevnice (*softline*), terénní hrany (*hardline*) a trigonometrické body (*points*). Posléze bylo z TINu odvozeno několik variant rastrových povrchů (o velikosti pixelu 2, 5 a 10 metrů). K tomuto účelu byla použita funkce *TIN to Raster*, byla vybrána lineární metoda výpočtu a velikost buňky (*Cellsize*) byla nastavena na odpovídající hodnoty velikosti pixelu. Srovnání velikostí VRML souborů vytvořených z uvedených rastrů je uvedeno v tab. 3. VRML soubor obsahující původní model terénu TIN byl příliš veliký a nebylo jej možné otevřít ani v desktopových prohlížečích *Instant Player* či *View3DScene*, natož jej dále zpracovávat. Na základě téměř stejné velikosti modelů terénu uložených pomocí tagu *ElevationGrid* byl pro další zpracování vybrán rastr s rozlišením 5 metrů.

Tab. 3. Srovnání velikostí (v kilobajtech) modelů terénu ve formátu VRML exportovaných z programu *ArcScene*

element pro uložení geometrie	originální TIN	rastr (2m)	rastr (5m)	rastr (10m)
<i>IndexedFaceSet</i>	37803,481	25370,883	25591,069	26188,465
<i>ElevationGrid</i>	-	1921,423	1921,375	1912,585

Pro transformaci dat z formátu VRML do X3DOM, respektive X3D, byl použit volně dostupný program *View3DScene*. Velikost dvou souborů, které vznikly po transformaci do X3D, jsou uvedeny v tab. 4. Dílčí úpravy (jako např. změny odkazů na rastrové soubory textur pokrývajících model terén) byly pro konverzi do X3D formátu prováděny v programu *Notepad ++*.

Při úpravě dat do podoby, jež lze zobrazit ve 3D scéně pomocí elementu *BVHRefiner*, bylo rovněž využito prostředí *ArcGIS* verze 10. Pro další zpracování byla vybrána rastrová vrstva o rozlišení 2, a ta byla exportována do formátu TIFF, jenž je vhodný pro další zpracování.

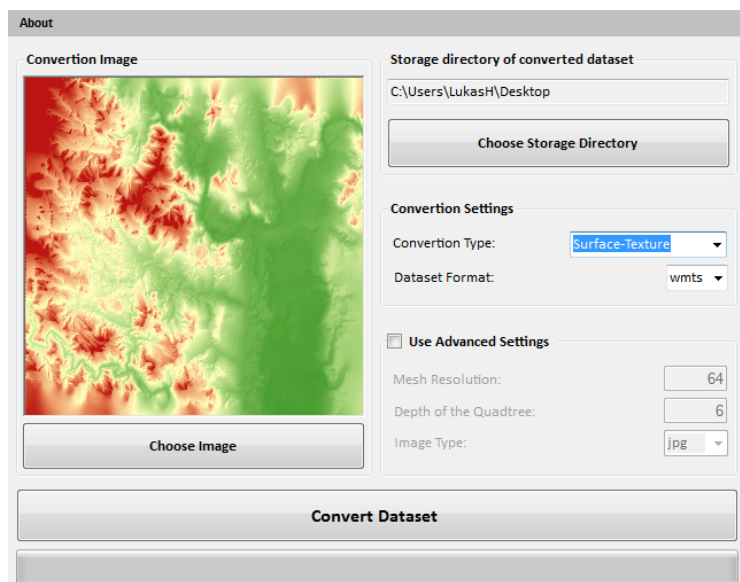
Kompresie dat v AOPT

Kompresi X3D souborů lze provádět pomocí nástroje *AOPT (Avalon-Optimizer)*, jenž je součástí softwarového balíku *Instant Reality*. Celý programový balík je dostupný zdarma, nevýhodou nástroje *AOPT* je pouze skutečnost, že jako ovládací rozhraní používá příkazovou řádku. Níže jsou přiloženy příkazy pro vytvoření binárního uložení, binárního uložení se ZIP kompresí a souborů s geometrií uloženou v elementech *BinaryGeometry* a *ImageGeometry*. Pro použití posledních dvou zmíněných způsobů je nutné nejdříve vytvořit složku, kam se v průběhu zpracování ukládají binární nebo rastrové soubory, dále je možné nastavit parametry těchto souborů. Např. výraz „sacp“ použitý níže by dle dokumentace k nástroji *AOPT* dostupné na X3DOM (2014) měl vytvářet nejkompaktnější výsledek.

```
aopt -i [input.x3d] -b [output].x3db
aopt -i [input.x3d] -z -b [output].x3db
mkdir bingeo
aopt -i [input.x3d] -G "bingeo/:sacp" -x [output].x3d
mkdir imggeo
aopt -i [input.x3d] -g "imggeo/:is" -x [output].x3d
```

Tvorba dlaždic pro dynamické načítání

Pro tvorbu dlaždicových struktur slouží open-source (licence GNU-GPL) program *BVHRefiner Dataset Converter*, jejímž autorem je Michael Englert z Fraunhofer IGD. Zatím je k dispozici pouze testovací verze aplikace (1.0), ve které nelze zpracovat rastrová data o libovolné velikosti hierarchické struktury (WMTS nebo TREE).



Obr. 4. Rozhraní aplikace *BVHRefiner Dataset Converter*

SROVNÁNÍ JEDNOTLIVÝCH METOD

ZIP komprese, binární uložení geometrie nebo využití *ImageGeometry* sice nekoresponduje se základním principem DOM, ale vzhledem k tomu že většina objemu X3D souborů je zpravidla tvořena

nestrukturovanými daty (seznamem souřadnic), nebrání tento rozpor v jejich použití. Úspora dat převažuje v tomto případě nad transparentností datové struktury.

Tab. 4. Srovnání komprese X3D souborů pomocí binárního uložení dat

	pův. soubor	binary		binary zipped	
	X3D	X3DB		X3DB	
	kilobajty	kilobajty	%	kilobajty	%
model terénu - <i>ElevationGrid</i>	1353,926	683,310	50,47	339,096	25,05
model terénu - <i>IndexedFaceSet</i>	33989,201	10086,507	29,68	2439,138	7,18

Binární uložení dat zmenšuje velikost modelů terénu asi na 50 a 30% velikosti původních souborů. Při použití ZIP komprese je snížení velikosti ještě výraznější (25 a 7% původní velikosti). Jak je zmíněno výše, obě varianty však není možné použít pro zobrazení dat prostřednictvím knihovny X3DOM.

Kompresi pomocí elementu *BinaryGeometry*, respektive *ImageGeometry* nelze použít na data uložená pomocí elementů *ElevationGrid*. Dojde pouze k úpravě (a mírnému snížení velikosti) X3D souboru, ale binární soubory (s příponou *bin* nebo *png*) se v programu AOPT nevytvoří. Vhodnějším elementem vstupní geometrie je *IndexedFaceSet*. K výraznému zmenšení velikosti dochází zejména v případě, že vstupní data jsou tvořena nižším počtem geometrických elementů obsahujících dlouhé seznamy bodů – souřadnic. Toto je právě případ modelů terénu (výsledná data mají 16 a 5% původního souboru). Hlavní slabinou použití elementů *ImageGeometry* a *BinaryGeometry* je skutečnost, že nemusí být správně vykreslovány – kupříkladu se výrazně liší zobrazení v desktopovém programu Instant Player a pomocí knihovny X3DOM.

Tab. 5. Srovnání komprese X3D dat pomocí elementů *BinaryGeometry* a *ImageGeometry*

	pův. soubor	BinaryGeometry				ImageGeometry			
	X3D	X3D	bin	celkem		X3D	img	celkem	
	kilobajty	kilobajty	kilobajty	kilobajty	%	kilobajty	kilobajty	kilobajty	%
model terénu - <i>ElevationGrid</i>	1353,926	1353,913	-	1353,913	100,00	1353,913	-	1353,913	100,00
model terénu - <i>IndexedFaceSet</i>	33989,201	34,942	5561,952	5596,894	16,47	48,756	1650,191	1698,947	5,00

Je nutné ještě doplnit, že ve výše uvedeném srovnání nebyla brána do úvahy velikost rastrových textur. Když 3D model obsahuje textury, je možné také snížit jejich velikost a zvýšit rychlost vykreslování scény úpravami textur. Konkrétně lze velikost textur zmenšit snížením rozlišení, nastavením vyšší úrovně ztrátové (soubory JPG) či bezztrátové (PNG) komprese. V případě elementu *BVHRefiner* jsou textury po zpracování uloženy do analogické struktury jako samotný model terénu. V ostatních případech jsou textury uloženy rovněž v externích souborech, na které je z X3D kódu odkazováno.

Výhodou v případě použití elementu *BVHRefiner* je také to, že lze zpracovávat data ve formátu TIFF, PNG, JPG či GIF. Modely terénu a textury lze vytvářet v celé řadě programů, a to nejen komerčních, ale i šířených zdarma či jako open-source. Ostatní způsoby komprese modelů terénu v X3DOM vyžadují téměř vždy jako první krok uložení modelu terénu do formátu VRML. Z tohoto výchozího otevřeného formátu lze teprve postupovat ve zpracování dále.

Naopak nesporným limitem aplikace *BVHRefiner Dataset Converter*, která je zatím jediným nástrojem, jenž podporuje automatizovanou tvorbu hierarchické struktury dlaždic, je skutečnost, že výsledné dlaždice mají vždy tvar čtverců.

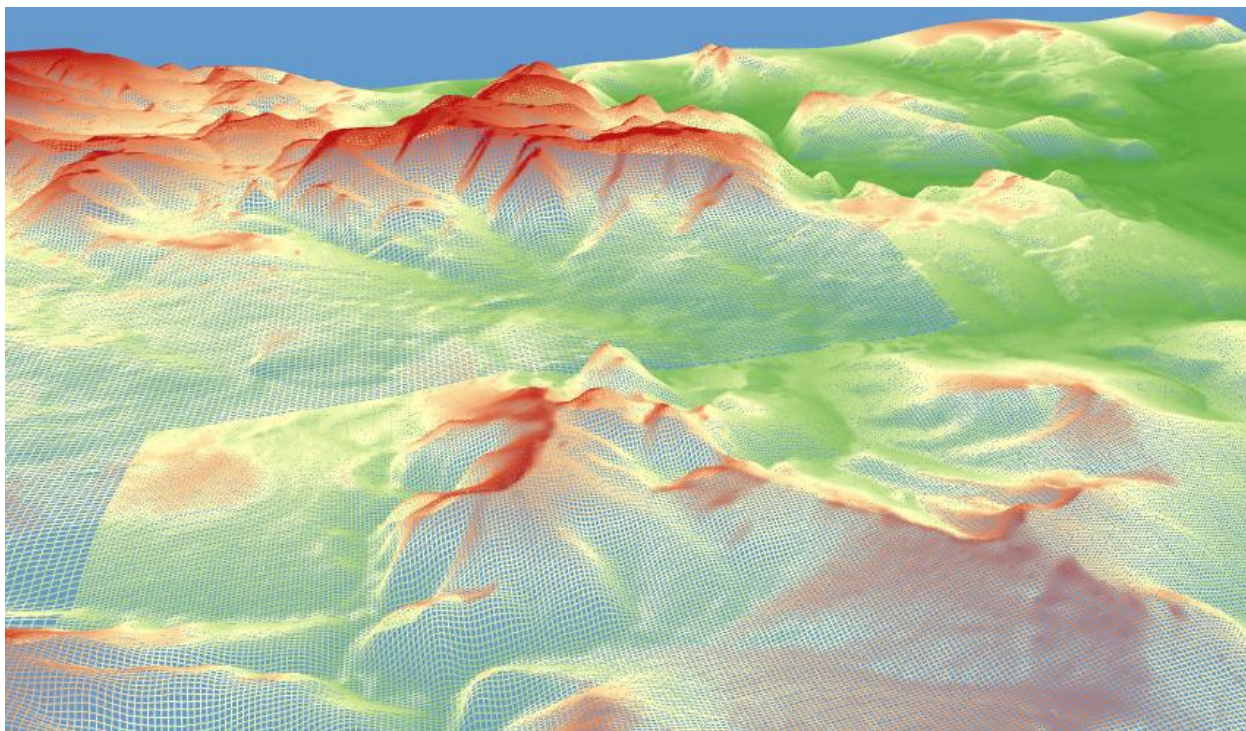
Tab. 6. Srovnání velikosti souborů v jednotlivých úrovních podrobnosti vytvořených v aplikaci *BVHRefiner Dataset Converter* (ve výše popsané struktuře WMTS)

úroveň	počet souborů	velikost (kilobajty)
0	1	20,732
1	4	67,985
2	16	203,472
3	64	558,056
4	256	1440,113
5	1024	3976,095
6	4096	10591,634

ZÁVĚR

Uložení dat digitálního modelu terénu umožňující jeho následné zobrazení pomocí knihovny X3DOM ve webovém prohlížeči je možné několika způsoby. V tomto příspěvku bylo testováno několik variant. Pro začátečníka nemusí být snadné se v nich zorientovat. Smyslem tohoto příspěvku bylo poskytnout základní přehled jednotlivých způsobů komprese modelů terénu. Díky nim je možné ve webovém prohlížeči zobrazovat i relativně rozsáhlé a podrobné modely terénu.

Zatímco načtení binárně uložených celých souborů X3D není do webového prohlížeče možné, pro modely terénu uložené jako TIN se ukázalo jako relativně úsporný způsob jejich uložení pomocí elementů *BinaryGeometry* a *ImageGeometry*. Limper et al. (2013) uvádí, že jak *BinaryGeometry*, tak *ImageGeometry* by měly být v X3DOM do budoucna nahrazeny elementem *ExternalGeometry*, která odkazuje na tzv. *POP Buffer*. Vývoj tohoto prvku knihovny X3DOM a jeho principy popisuje Limper et al. (2014). Pro rastrová data lze doporučit uložení pomocí elementu *ElevationGrid* v případě menších nebo málo podrobných modelů. V případě rozsáhlejších a více podrobných rastrových modelů terénu se jeví jako výhodnější variant jejich uložení pomocí elementu *BVHRefiner*.



Obr. 5. Postupné načítání jednotlivých dlaždic v rámci elementu *BVHRefiner*

Skutečností, které poněkud komplikuje tvorbu webových 3D vizualizací modelů terénu je fakt, že pro zpracování dat je nezbytné využívat různé softwarové nástroje. Je nutné používat rozličné programy pro různé kroky zpracování. V tomto příspěvku byl pro zpracování dat využit software *ArcGIS*, nástroj *AOPT*,

programy *BVHRefiner Dataset Converter* a *Notepad++*. V praxi je však možné využít i celou řadu dalších programů, a to jak programy komerční (př. *FME*), tak celou řadu volně dostupných či dokonce open source programů (např. *3DEM*, *Blender*, *LandSerf*, *MeshLab*, *View3DScene*, *VTBuilder* nebo *Wings 3D*). To, že lze využít řadu programů by se mohlo jevit jako výhoda, avšak každá z výše zmíněných aplikací vytváří modely terénu více či méně odlišně. Do budoucna by bylo žádoucí, kdyby docházelo ke sjednocování a rozšiřování interoperability. Lze rovněž předpokládat, že jednotlivé nástroje, zejména ty určené pro kompresi 3D souborů, budou dále optimalizovány. U některých, jako je *BVHRefiner Dataset Converter*, se toto očekává již v blízké budoucnosti.

LITERATURA

- Behr, J., Eschler, P., Jung, Y., Zöllner, M. (2009) X3DOM – A DOM-based HTML5/ X3D Integration Model, [on-line]. <<http://www.web3d.org/x3d/wiki/images/3/30/X3dom-web3d2009-paper.pdf>> [cit. 1. 12. 2014].
- Behr, J., Jung, Y., Drevensek, T., Aderhold, A. (2011) Dynamic and Interactive Aspects of X3DOM, [on-line]. <<http://www.past.igd.fraunhofer.de/~yjung/pubs/11x3dom.pdf>> [cit. 2. 12. 2014].
- Behr, J., Jung, Y., Keil, J., T. Drevensek, T., Zöllner, M., Eschler, P., Fellner, D. (2010) A Scalable Architecture for the HTML5/ X3D Integration Model X3DOM, [on-line]. <<http://www.x3dom.org/x3dom/doc/paper/2010-x3dom-web3d.pdf>> [cit. 3. 12. 2014].
- Brown, T. B., Butters, K., Panda, S. (2014) *HTML5 Okamžitě*. 1. vyd. Computer Press, Brno. 256 s. ISBN 978-80-251-4296-7.
- Herman, L. (2013) 3D kartografická vizualizace v prostředí internetu: technologické možnosti a bariéry. In Osman, R (ed.), *Geografický výzkum: Společnost a příroda v období krize*. Masarykova univerzita, Brno, s. 45-55. ISBN 978-80-210-6230-6.
- Herman, L., Řezník, T. (2013) Web 3D Visualization of Noise Mapping for Extended INSPIRE Buildings Model. In Hřebíček, J., Schimak, G., Rizzolli, A., Kubásek, M. (eds.), *Environmental Software Systems. Fostering sharing information*. Springer, Heidelberg. s. 414-424. ISBN 978-3-642-41150-2.
- Lienert, C., Jenny, B., Schnabel, O., Hurni, L. (2012) Current Trends in Vector-Based Internet Mapping: A Technical Review. In Peterson, M. P. (ed.), *Online Maps with APIs and WebServices*. Springer, Berlin. s. 23-36. ISBN 978-3-642-27484-8.
- Limper, M., Jung, Y., Behr, J., Alexa, M. (2013) The POP Buffer: Rapid Progressive Clustering by Geometry Quantization [on-line]. <<http://x3dom.org/pop/files/popbuffer2013.pdf>>. [cit. 10. 12. 2014].
- Limper, M., Thöner, M., Behr, J., Fellner, D. W. (2014) SRC – A Streamable Format for Generalized Web-based 3D Data Transmission [on-line]. <http://x3dom.org/src/files/Web3d2014_SRC.pdf> [cit. 10. 12. 2014].
- Voženílek, V. (2005) *Cartography for GIS: Geovisualization and Map Communication*. 1st ed. Univerzita Palackého v Olomouci, Olomouc. 142 s. ISBN 8024410478.
- Wood, J., Kirschenbauer, S., Döllner, J., Lopes, A., Bodum, L. (2005) Using 3D Visualization. In Dykes, J., MacEachren, A. M., Kraak, M. J. (eds.) *Exploring Geovisualization*, Elsevier Ltd., Kidlington, s. 295-312, ISBN 0-08-044531-4.
- X3DOM (2014) URL: <<http://www.x3dom.org/>> [cit. 15-12-2014].