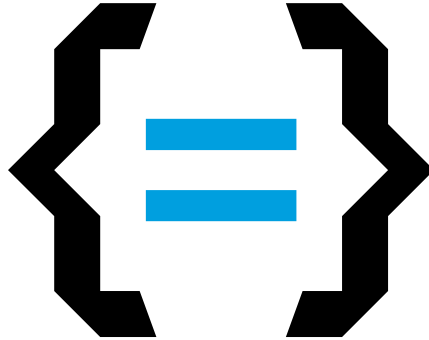


**Univerzita Hradec Králové**  
**Fakulta informatiky a managementu**  
**Katedra informatiky a kvantitativních metod**



Webová interaktivní vizualizace georeferencovaného 3D modelu

Web interactive visualization of georeferenced 3D model

Bakalářská práce

Autor: Mgr. Rostislav Berka  
Studijní obor: Aplikovaná informatika

Vedoucí práce: Ing. Bruno Ježek, Ph.D.

Prohlašuji, že jsem bakalářskou práci zpracoval samostatně a s použitím uvedené literatury.

V Hradci Králové dne

Rostislav Berka

Rád bych poděkoval vedoucímu bakalářské práce Ing. Bruno Ježkovi, Ph.D. za metodické vedení práce, vstřícný přístup a cenné rady poskytnuté v průběhu zpracování. Dále bych chtěl poděkovat Spolku Přátelé Zahrádky, jmenovitě Ing. Janu Čihákovi, který mi poskytl velké množství fotografií, s jejichž pomocí bylo možno vytvořit 3D model Zahrádky.



## **Anotace**

Bakalářská práce se zabývá problematikou zobrazení detailních georeferencovaných 3D modelů území zahrnujících digitální model terénu i budovy ve webovém prostředí. V práci jsou rozebrány charakteristiky a specifika vizualizace georeferencovaných 3D modelů na webu, jako je problematika 3D dlaždic, používané souřadnicové systémy, formáty a knihovny. Pro praktickou implementaci byla použita knihovna CesiumJS. Aplikace vizualizuje i vektorová data (body zájmu). Nakonec jsou diskutovány výhody a nevýhody použitého řešení. Výsledkem je komplexní webová aplikace naprogramovaná s pomocí knihovny CesiumJS umožňující uživateli prohlížet georeferencovaný 3D model dané oblasti ve třech časových obdobích (zaniklá obec Zahrádka u Ledče nad Sázavou).

**Klíčová slova:** georeferencovaný 3D model, 3D dlaždice, CesiumJS, JavaScript, Zahrádka u Ledče nad Sázavou

## **Annotation**

**Title: Web interactive visualization of georeferenced 3D model**

The bachelor thesis deals with the problem of visualization of detailed georeferenced 3D models of the sites including digital terrain model and buildings in a web environment. The thesis discusses the characteristics and specifics of the visualization of georeferenced 3D models on the web, such as 3D tiles, coordinate systems, formats and libraries. CesiumJS was used for practical implementation. Application also visualises vector data (points of interest). Finally, the advantages and disadvantages of the solution used are discussed. The result is a complex web application programmed with the CesiumJS library allowing the user to explore a georeferenced 3D model of a specific area in three time slices (the lost town of Zahrádka u Ledče nad Sázavou).

Key words: georeferenced 3D model, 3D tiles, CesiumJS, JavaScript, Zahrádka u Ledče nad Sázavou

## Obsah

1	Úvod, motivace.....	1
2	Teoretická část.....	3
2.1	Digitální modely terénu.....	3
2.2	Knihovny pro vizualizaci georeferencovaných 3D dat na webu .....	6
2.2.1	Google Earth API – již ukončeno .....	6
2.2.2	CesiumJS .....	7
2.2.3	deck.gl.....	8
2.3	Webové mapové služby.....	9
2.4	3D Tiles.....	10
2.5	Vybrané datové formáty v GIS .....	11
2.5.1	2D vektorové formáty .....	11
2.5.2	3D formáty .....	12
2.6	Vybrané souřadnicové systémy v GIS a kartografii.....	13
2.6.1	EPSG kódy .....	14
2.6.2	System jednotné trigonometrické sítě katastrální (S-JTSK).....	15
2.6.3	WGS-84 Universal Transverse Mercator (UTM).....	15
2.6.4	Web Mercator .....	15
3	Metodika zpracování.....	17
3.1	Charakteristika zájmového území.....	17
3.2	Příprava dat .....	18
3.3	Implementace v CesiumJS .....	20
3.3.1	Inicializace 3D okna .....	20
3.3.2	Přidání podkladů .....	21
3.3.3	Vlastní uživatelské rozhraní a obsluha událostí.....	24
4	Testování a výsledky.....	27

4.1	Měření FPS.....	27
4.2	Výsledná implementace a 3D vizualizace.....	28
5	Závěr.....	33
6	Seznam použité literatury.....	34
7	Přílohy .....	36

## Seznam obrázků

Obrázek 1: Náměstí v Zahrádce v zimě – render, vlastní tvorba.....	1
Obrázek 2: DMT se stínováním terénu a hladinou v. n. Švihov v Zahrádce, vlastní tvorba .....	3
Obrázek 3: Metody sběru dat pro tvorbu DMT [3] .....	4
Obrázek 4: Triangulace [4].....	5
Obrázek 5: 3D vizualizace náměstí v Užhorodě [6].....	7
Obrázek 6: Platforma Cesium [7].....	8
Obrázek 7: Princip dlaždicování [12].....	10
Obrázek 8: Využití adaptivního quadtree v oblasti s nerovnoměrně rozmístěnou zástavbou (Cambridge) [7] .....	11
Obrázek 9: Využití COLLADA a glTF v různých SW [14] .....	13
Obrázek 10: Válcové zobrazení [16].....	14
Obrázek 11: Tissotova indikatrix – Mercatorovo zobrazení [17] .....	16
Obrázek 12: Poloha obce Zahrádka [1] .....	18
Obrázek 13: Georeferencování 3D modelu v Cesium Ion, vlastní tvorba .....	20
Obrázek 14: Náměstí se stíny/bez stínů, vlastní tvorba .....	21
Obrázek 15: PinBuilder, vlastní tvorba .....	22
Obrázek 16: Vizualizace otvoru v Cesium DMT pro 3D model Zahrádky, vlastní tvorba .....	23
Obrázek 17: Výběr podkladů ČÚZK a mapy.cz, vlastní tvorba .....	24
Obrázek 18: Vlastní vs Defaultní pop up, vlastní tvorba .....	25
Obrázek 19: Popis aplikace, vlastní tvorba.....	28
Obrázek 20: Blender vs CesiumJS, vlastní tvorba.....	29
Obrázek 21: Nekvalitní DMT poskytovaný Cesiumem, vlastní tvorba .....	30
Obrázek 22: Budovy neodrážející světlo (vpravo), vlastní tvorba .....	32

## Seznam tabulek

Tabulka 1: Specifikace PC.....	27
Tabulka 2: Počty geometrických primitiv modelu.....	31





# 1 Úvod, motivace

Předkládaná bakalářská práce se věnuje vizualizaci detailního georeferencovaného 3D modelu krajiny ve webovém prostředí. 3D modelem krajiny se rozumí digitální model terénu (DMT) s usazenými budovami. V uplynulých letech se stalo běžným standardem 2D zobrazení zemského povrchu v interaktivních mapových aplikacích na webu či mobilních zařízeních. S rostoucím výkonem HW a rozvojem technologií a metod jako je Structure from Motion fotogrammetrie, laserové skenování (LiDAR), případně i ruční 3D modelace se dostává do popředí webová 3D vizualizace. V práci je implementována webová single page aplikace umožňující uživateli prohlížení 3D modelu zaniklé obce Zahrádka u Ledče nad Sázavou (Obrázek 1). Scéna obsahuje taktéž interaktivní značky, které se vztahují k bodům zájmu. Aplikace je připravená pro specifika vizualizace velkých 3D modelů.



**Obrázek 1: Náměstí v Zahrádce v zimě – render, vlastní tvorba**

V práci je využita cloudová platforma Cesium Ion a open-source JavaScriptová knihovna CesiumJS založená na WebGL. Autor se dlouhodobě věnuje zmíněnému zájmovému území a v minulosti vznikla například interaktivní mapa obce nebo atlas

zátopové oblasti Švihova. V aplikaci je prezentován 3D model zaniklé obce Zahrádka u Ledče nad Sázavou, jehož tvorba není předmětem zájmu této práce, neboť prvních několik budov vzniklo v rámci staršího autorova projektu. [1] Drtivá většina objektů však byla vymodelována ve volném čase autora.

Cílem práce je prozkoumat problematiku zobrazení detailních georeferencovaných 3D modelů území zahrnujících digitální model terénu i budovy ve webovém prostředí a následně s pomocí zvolené knihovny implementovat webovou interaktivní 3D aplikaci, která umožní uživateli prohlížet 3D model dané oblasti.

## 2 Teoretická část

### 2.1 Digitální modely terénu

Digitální model terénu, respektive reliéfu (DMT, DMR) představuje reprezentaci zemského povrchu bez vegetace a antropogenních útvarů (Obrázek 2). Příbuzný je pak digitální model povrchu (DMP), který na rozdíl od DMT zahrnuje právě i objekty na zemském reliéfu přírodního i antropogenního původu. DMT i DMP nacházejí využití v nejrůznějších geoinformačních analýzách a vizualizacích. [2]



**Obrázek 2: DMT se stínováním terénu a hladinou v. n. Švihov v Zahrádce, vlastní tvorba**

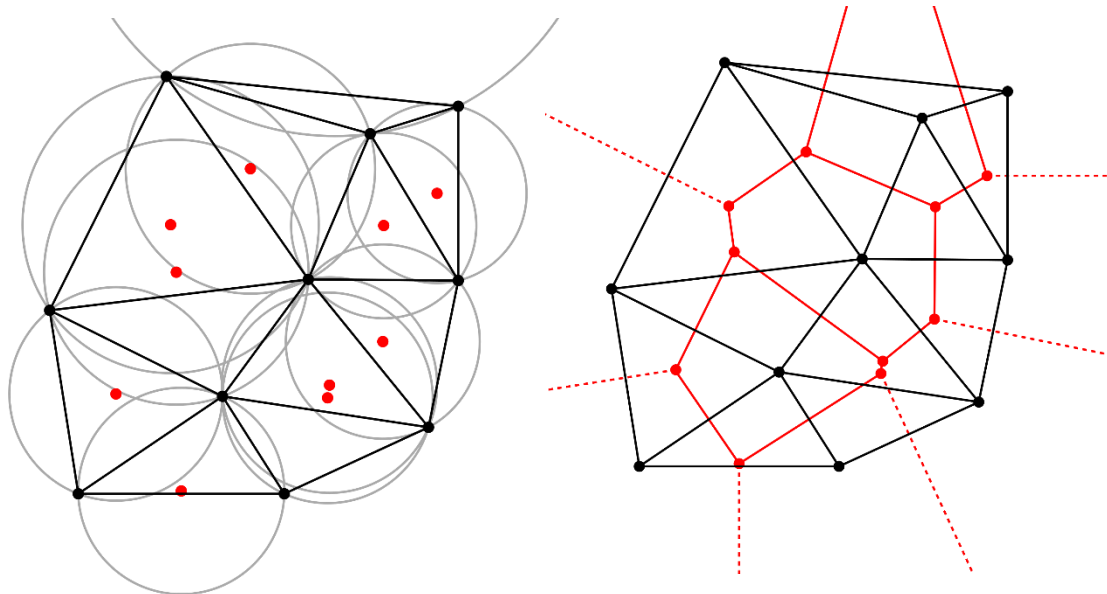
Zdrojová data pro tvorbu DMT a DMP mohou být získána buď měřením kontaktním či distančním. Kontaktním měřením rozumíme například geodetické měření jako je nivelace a tachymetrie. V případě distančního měření hovoříme o metodách jako fotogrammetrie, laserové skenování terénu nebo radarové snímání (Obrázek 3). Druhotně pak můžeme získat podkladová data vektorizací vrstevnic tištěných map. [2]

Metoda	přesnost	Rychlost	Cena	Velikost území
Nivelace	0,001 – 0,01 m	*	€€€€€€€	o
Tachymetrie	0,02 – 0,05 m	**	€€€€€	o
GPS	0,02 – 0,05 m	***	€€€€	o
UAV	0,5 – 0,01 m	*****	€	oo
Fotogrammetrie	0,2 – 0,5 m	*****	€€	O
DPZ	1 – 10 m	*****	€	OO
InSAR	5 m změny 1 cm	*****	€	OO
Radarové snímání	20 – 70 m	*****	€	OO
LIDAR	0,1 – 0,2 m	*****	€€€	O
manuální vektorizace map	dle měřítka mapy	**	€€€	libovolně
automatická vektorizace map	dle měřítka mapy	****	€	libovolně

**Obrázek 3: Metody sběru dat pro tvorbu DMT [3]**

Všemi výše uvedenými metodami však získáme pouze bodová diskrétní měření. Pro vytvoření spojitě reprezentace povrchu musíme použít konkrétní interpolační metodu pro dopočet nadmořských výšek i v místech, kde nebyla měřena. [2]

Existují dva základní typy digitálních modelů terénu. Prvním jsou polyedrické DMT, přičemž nejčastěji se jedná o TIN (triangulated irregular network). Druhým případem jsou rastrové DMT. TIN tvoří jednotlivé výškové body, které jsou propojené hranami. TIN vzniká triangulací, přičemž v oblasti geoinformatiky je běžné využívat Delauneho triangulaci (Obrázek 4). Zkonstruování triangulace je poměrně jednoduché. Třemi body se proloží kružnice. Pokud uvnitř kružnice leží další bod, zvolí se jiné tři body. V opačném případě je z těchto třech bodů vytvořen trojúhelník triangulace. Tento postup zajišťuje, že se výsledné trojúhelníky triangulace maximálně blíží rovnostranným trojúhelníkům. Triangulace splňující tuto podmínku pak nejlépe vystihuje terén. Delauneho triangulace je jednoznačná, výsledná triangulace je nezávislá na volbě počátečního bodu.



*Černě: datové body s triangulací, červeně: středy kružnic opsaných jednotlivým trojúhelníkům. Vpravo: středy kružnic tvoří vrcholy tzv. thiessenových polygonů (resp. Voroného diagramy či teselace)*

**Obrázek 4: Triangulace [4]**

Výhodou TIN je adaptabilita na lokální terén. V plochých oblastech jako jsou nížiny je pouze malé množství velkých trojúhelníků. V oblastech s variabilním terénem (hory) je umístěno větší množství menších trojúhelníků pro co možná nejlepší aproximaci průběhu terénu. Toto vede k lepší datové optimalizaci. Výhodou TIN je možnost použít tzv. povinné hrany (breaklines). Povinnými hranami je možno ovlivnit jinak jednoznačný průběh triangulace a ve vzniklém DMT namodelovat nejrůznější terénní útvary a hrany (např. koryta toků, meze, břehovky vodních ploch, skalní hrany aj.). Nevýhodou TIN je možnost vzniku vodorovných trojúhelníků, a to například v místech vrcholů kopců nebo v údolích vodních toků. Tento jev je nežádoucí, neboť TIN v těchto místech nereflktuje přirozený průběh terénu. [3]

Druhou reprezentací je rastr. Rastr je tvořen pravidelnou sítí buněk (pixelů) s určitým prostorovým rozlišením. Toto rozlišení udává, jak velkou část zemského povrchu v realitě reprezentuje jeden pixel rastru. Pixely jsou nejčastěji čtvercové. Topologie buněk (sousedství) je jednoznačně dáno rastrovou mřížkou. Mřížka bývá

obvykle orientována stejně s použitým kartézským souřadnicovým systémem. Rastr je na rozdíl od TIN náchylný k nadhodnocení či podhodnocení (overestimation, underestimation) při aproximaci terénu v oblastech s proměnlivou členitostí terénu (např. příliš mnoho malých pixelů v nížině). Rastrový DMT vzniká interpolací z diskrétních bodových měření. Existuje mnoho různých interpolačních metod, příkladem může být IDW (inverse distance weighting), kriging, natural neighbour, spline, či proprietární algoritmus Topo to raster firmy Esri. TIN také může být převeden na rastr. [3] Existuje souvislost mezi Delaunayho triangulací a interpolací natural neighbour. Při této interpolaci se prostor kolem datových bodů rozdělí na tzv. Voroného diagramy. Vrcholy těchto diagramů (polygonů) jsou potom středy kružnic opsaných trojúhelníkům triangulace.

V současnosti běžně používané DMT v geoinformatice jsou pouze 2,5D nikoliv 3D, jelikož pro každý XY bod existuje pouze jedna nadmořská výška Z. Nelze tak modelovat například terénní a skalní převisy, skalní brány apod.

## ***2.2 Knihovny pro vizualizaci georeferencovaných 3D dat na webu***

### **2.2.1 Google Earth API – již ukončeno**

V minulosti bylo k dispozici pro vizualizaci geografických 3D dat na webu Google Earth API. Toto API bylo možno využít na vlastních stránkách. API umožňovalo na virtuální glóbus přidávat objekty jako například mapové značky, polygony, linie ve formátu KML a samozřejmě 3D modely. Pro jejich tvorbu Google poskytoval open-source SW SketchUp. Bohužel postupem času zastarala technologie, na které bylo API postaveno (NPAPI plugin framework) a Google ukončil podporu API v roce 2015. [5] Taktéž SketchUp koupila společnost Trimble a vytvořila z něj komerční proprietární řešení pro architekty. Google Earth API bylo použito například pro 3D rekonstrukci a vizualizaci náměstí v Užhorodě. [6]





**Obrázek 5: 3D vizualizace náměstí v Užhorodě [6]**

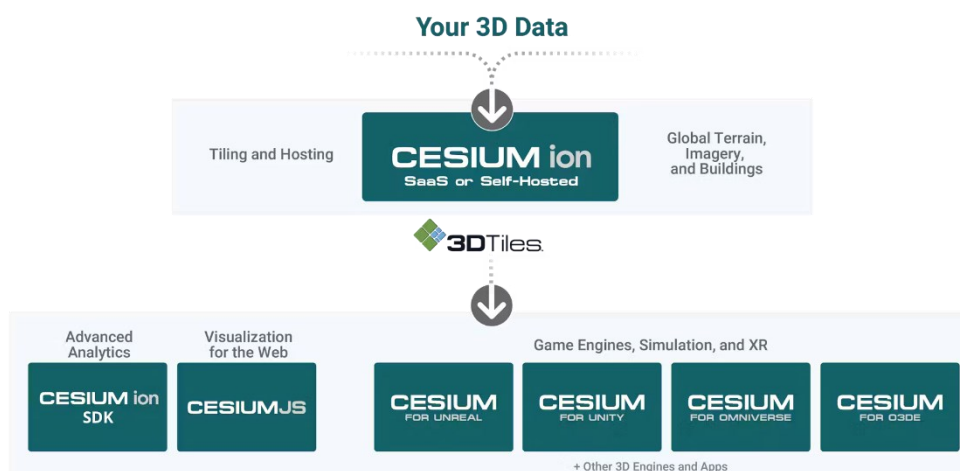
### 2.2.2 CesiumJS

Cesium je platforma určená pro vizualizaci georeferencovaných dat ve vlastních aplikacích. Ekosystém zahrnuje cloudové řešení pro sdílení vlastních 3D modelů (Cesium Ion). Podporovány jsou nejrůznější typy 2D i 3D dat jako jsou například bodová mračna, produkty laserového skenování a fotogrammetrie, BIM a CAD modely, ruční modelace z 3D SW jako je například Blender, rastrové DMT, rastrová obrazová data, vektorová data aj. Cloudová služba dokáže z nahraných 3D modelů vytvořit 3D dlaždice splňující standard 3D Tiles. Právě s pomocí 3D Tiles lze 3D modely načítat do nejrůznějších aplikací. Ve webovém prostředí Cesium Ion je možné nahrané 3D modely georeferencovat. V Cesium Ion jsou k dispozici také podklady Google (3D dlaždice pro celý svět vytvořené metodami fotogrammetrie), 3D budovy OpenStreetMap, obrazová data z družice Sentinel-2 pro celý svět, mapové podklady Microsoftu (Bing maps) a další. Tyto podklady je možné načítat do vlastních aplikací a zobrazovat je na virtuálním 3D glóbu. [7]



Další součástí Cesium ekosystému je CesiumJS. Jedná se o JavaScriptovou knihovnu pro tvorbu 3D map a prezentování 3D modelů na webu. Knihovna podporuje rozsáhlé 3D modely kompatibilní se standardem 3D Tiles. Podporována jsou vektorová i rastrová data. CesiumJS umí pracovat i s časem ve scéně, ve výchozím stavu v aplikaci je přítomna časová osa. Pro časové dynamické animace byl Cesiumem vytvořen formát CZML. Knihovna využívá WebGL2, při renderování je využita grafická karta, což má pozitivní dopad na výkon aplikace. [7]

Dále Cesium poskytuje pluginy pro nejznámější herní enginey jakým je Unity Engine a Unreal Engine, O3DE engine a další. [7] CesiumJS bylo použito například pro dokumentaci vývoje krajiny v katastrálním území zaniklé obce Holešice (Mostecko) zasažené povrchovou těžbou uhlí. [8] Z dalších 3D aplikací prezentujících vývoj využití krajiny na vybraných územích lze zmínit například výstup projektu NAKI – Zaniklé krajiny Česka (interaktivní mapová aplikace). [9]



**Obrázek 6: Platforma Cesium [7]**

### 2.2.3 deck.gl

Konkurencí pro CesiumJS může být framework deck.gl, který slouží pro prezentaci geografických 2D i 3D dat na webu. Framework je taktéž založen na WebGL2. Má

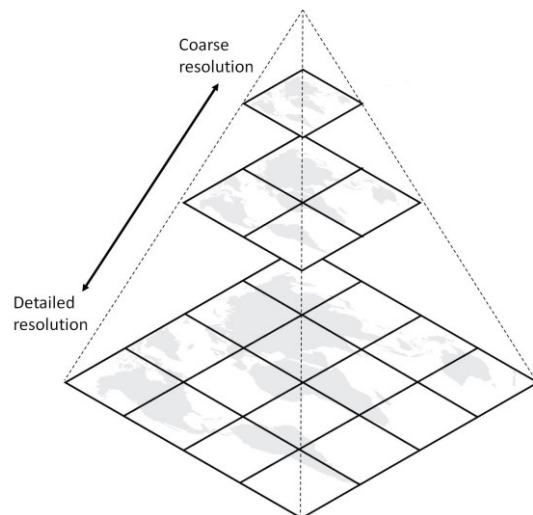
dobrou integraci s frameworkem React a dalšími. Nabízí mnoho různých podkladových map, jedná se například o podklady Google Maps, Mapbox, Esri, MapLibre a další. Deck.gl podporuje nejrozličnější formáty jako KML, GeoJSON, CSV, pro 3D data pak například gLTF. Umožňuje také načítat 3D modely a vrstvy z cloudu Cesium Ion. [10]

## **2.3 Webové mapové služby**

Pro distribuci mapových podkladů přes internet slouží v oblasti geoinformatiky webové mapové služby. Tyto služby jsou standardizovány mezinárodní organizací Open Geospatial Consortium. Členy jsou organizace z veřejného (NASA, ESA, USGS a další) i soukromého (Google, Esri, Amazon, Oracle apod.) sektoru. Existuje více než 450 členských organizací. [11]

Nejznámější webovou mapovou službou je Web Map Service (WMS). V rámci požadavku na službu je odeslána definice vrstvy a oblast zájmu s pomocí souřadnic bounding boxu požadovaného výřezu. Služba následně vrací samotný výřez ve formátu JPEG, PNG apod. [11]

Sofistikovanější službou je Web Map Tile Service (WMTS). Jak již z názvu vyplývá, základem je mapový podklad rozdělený na bezešvé dlaždice. Tyto dlaždice mají různou úroveň podrobnosti. Na základě přiblížení (zoomu) mapového podkladu jsou distribuovány dlaždice v příslušném rozlišení (měřítku). Dlaždic je běžně i více než 20 úrovní. S každou další úrovní se zčtyřnásobuje počet dlaždic (Obrázek 7). Při posunutí mapového výřezu je nutné dodat již pouze chybějící dlaždice, což vede k optimalizaci výkonu aplikace a nižší náročnosti na přenesená data. [12] Existují i další služby jako WFS, WCS aj.



**Obrázek 7: Princip dlaždicování [12]**

## 2.4 3D Tiles

3D Tiles jsou navrženy pro sdílení a renderování rozsáhlých 3D modelů, jako jsou například výstupy fotogrammetrie, laserového skenování (bodová mračna), BIM/CAD, a ručních 3D modelací. [11] 3D Tiles jsou standardizovány Open Geospatial Consortium, za vznikem tohoto standardu stojí právě Cesium. 3D Tiles podporují formáty jako COLLADA, OBJ a glTF. Specifickou vlastností je Hierarchical Level of Detail (HLOD). Dlaždice na vrcholu hierarchie mají nízkou úroveň detailu. Jejich potomci mají úroveň detailu naopak vyšší. Při renderování je možno zvolit dlaždice s příslušnou úrovní detailu. Snahou je dosáhnout kompromisu mezi výkonem a kvalitou renderování. Pro zvolení dlaždice se správnou úrovní detailu se posuzuje geometrická chyba. Pouze dlaždice, které jsou zrovna viditelné, jsou renderovány. Základním elementem 3D Tiles je tileset popsáný s pomocí JSON souboru. Jedná se o hierarchickou datovou strukturu (strom). Může se jednat o k-d tree, quadtree (Obrázek 8), octree nebo grid. Základem je tedy root tile, která má child tiles. Jeden tileset může kombinovat různé formáty – například bodové mračno (Point Cloud) a ruční modelace (Batched 3D Model). Součástí tilesetu může být také například Instanced 3D model. Jedná se o jedinou geometrii, která se renderuje ve scéně vícekrát. Instanced 3D model nachází využití u opakujících se objektů jako

jsou například plaňky plotu. Batched 3D Model je většinou binární glTF. Skládá se z vertexů, normál (nepovinné) a namapované textury. Každá dlaždice má tzv. Bounding Volume, kterých je více typů. Slouží například pro testování překrytí (intersect). [7]



**Obrázek 8: Využití adaptivního quadtree v oblasti s nerovnoměrně rozmístěnou zástavbou (Cambridge) [7]**

## **2.5 Vybrané datové formáty v GIS**

### **2.5.1 2D vektorové formáty**

Jedním z běžně užívaných formátů v geoinformatice je KML (Keyhole Markup Language), který je založen na XML. Jedná se o formát pro uložení geografických dat včetně kartografické reprezentace. Jeho použití je typické pro webové aplikace.

Umožňuje uložení bodů, linií, polygonů. Samozřejmě je standardizován Open Geospatial Consortium. [11] Jeden soubor může obsahovat různé geometrie, a dokonce i rastrová data. Formát slouží hlavně pro vizualizaci dat spíše než pro analýzu. Typické je jeho použití v Google Earth. [13]

JSON (JavaScript Object Notation) je často užívaným formátem na webu pro přenos dat. V současnosti je postupně upřednostňován před XML. Jeho modifikací užívanou v GIS sféře je GeoJSON, který je standardizován OGC. GeoJSON může obsahovat objekty (body, linie, polygony aj.), přičemž je zahrnuta jak geometrie, tak atributy. Jedná se o dobře čitelný formát. Na rozdíl od KML nezahrnuje kartografickou reprezentaci. [11] CZML je formát založený na JSON, který umožňuje ukládat časově dynamická data primárně pro vizualizaci ve webových aplikacích. Standard vznikl pro účely vizualizace v CesiumJS. [7]

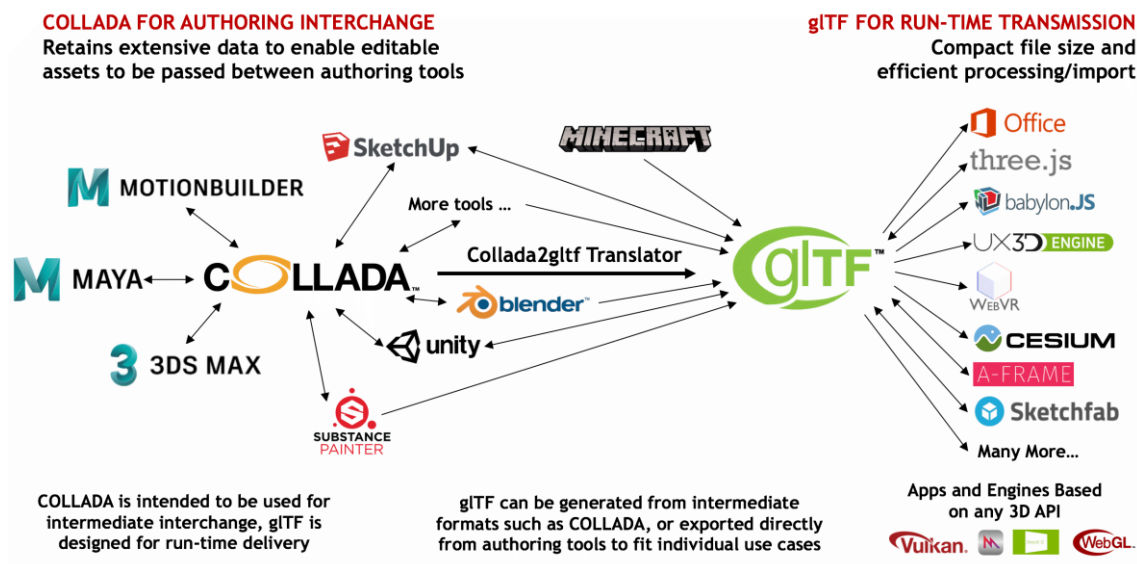
### **2.5.2 3D formáty**

Khronos Group je organizace s více než 180 členy, která vytváří standardy v oblasti 3D grafiky, VR, AR, machine learningu aj. Členy jsou mimo jiné největší firmy jako např. AMD, Apple, Google, Nvidia, Samsung, Sony a další. [14]

COLLADA je 3D formát založený na XML. Jedná se o starší formát dnes nahrazovaný glTF. Podporuje geolokaci objektů, proto se užívá právě v GIS sféře. Je kompatibilní s mnohými aplikacemi běžně užívanými v počítačové grafice. Podporuje také fyzikální simulace a animace. Nevýhodou může být větší velikost souboru. Původním autorem formátu je firma Sony. Collada jakožto standard v současnosti spadá pod Khronos Group. [14, 15]

glTF je formát pro 3D grafiku. Základem je JSON soubor popisující scénu obsahující 3D modely. Scéna může obsahovat síť (mesh), kamery, textury, animace a další. Existuje také Binary glTF (přípona .glb), který obsahuje veškerá data v jediném binárním souboru. glTF minimalizuje velikost výsledného 3D souboru. [14] 3D Tiles

jsou založeny právě na binárním glTF, přičemž využívají úspornosti zmíněného formátu. [7]



Obrázek 9: Využití COLLADA a glTF v různých SW [14]

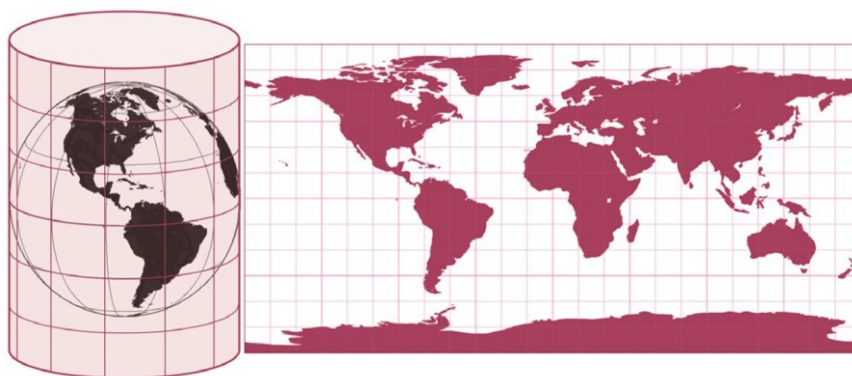
## 2.6 Vybrané souřadnicové systémy v GIS a kartografii

Kartografické zobrazení slouží pro převod objektů z referenční plochy do roviny mapy. Referenční plochou rozumíme těleso, kterým nahrazujeme Zemi při kartografických výpočtech. V praxi se jedná buď o referenční kouli ( $R = 6371,11 \text{ km}$ ) nebo elipsoid. Povrch těchto těles samozřejmě nelze rozvinout do roviny. Kartografické zobrazení využívá tzv. zobrazovací plochu pro převod do roviny mapy. Zobrazovací plocha může být buď válec (válcové zobrazení), kužel (kuželové zobrazení) nebo rovina (azimutální zobrazení). Kolem referenční plochy představující Zemi ovineme např. plášť válce (zobrazovací plocha). Poté můžeme přenášet obraz (objekty na Zemi) na plášť válce, který následně rozvineme do roviny mapy (Obrázek 10). Pro převod polohy objektů slouží zobrazovací rovnice, které jsou unikátní pro každé kartografické zobrazení. Jedná se o obecné funkce, které popisují závislost kartézských souřadnic v rovině mapy (X a Y) na

zeměpisných souřadnicích ( $\varphi$  – zem. šířka,  $\lambda$  – zem. délka). [16] Lze je vyjádřit ve tvaru:

$$X = f(\varphi, \lambda) \quad Y = g(\varphi, \lambda)$$

Převod kulové plochy do roviny samozřejmě produkuje zkreslení. Toto zkreslení může být délkové, úhlové a plošné, zkreslení je zároveň proměnlivé v závislosti na poloze v mapě a směru. Při volbě vhodného zobrazení pro daný účel musíme uvažovat toto zkreslení. Konkrétně lze například zmínit volbu zobrazení zachovávající úhly pro námořní navigaci. Průběh zkreslení v mapě lze znázornit s pomocí Tissotovy indikatrix. [16] Kartografický souřadnicový systém má typicky definovanou použitou referenční plochu a konkrétní kartografické zobrazení ve zvolené poloze. Jednotkami jsou metry. Zeměpisný souřadnicový systém má definovanou pouze referenční plochu a jednotkami jsou stupně.



**Obrázek 10: Válcové zobrazení [16]**

### 2.6.1 EPSG kódy

EPSG kódy slouží pro jednoznačné určení souřadnicového systému. Používají se pro zeměpisné i kartografické souřadnicové systémy. EPSG (European Petroleum Survey Group) byla organizace s přesahem do naftového průmyslu. Dataset EPSG kódů spravuje nástupnická organizace IOGP (The International Association of Oil & Gas producers). [17] Přehled souřadnicových systémů je k dispozici na [www.epsg.io](http://www.epsg.io).



### **2.6.2 Systém jednotné trigonometrické sítě katastrální (S-JTSK)**

Je geodetický souřadnicový systém používaný na území ČR (EPSG 5514), který vznikl ve 20. letech v ČSR. Je definovaný křovákovým zobrazením nad Besselovým elipsoidem. Jedná se o konformní kuželové zobrazení. Zkreslení na území státu je minimální (maximální hodnota 14 cm/1 km). Používá se v civilním sektoru např. v katastru nemovitostí, pro distribuci většiny dat ze strany ČÚZK aj. [16]

### **2.6.3 WGS-84 Universal Transverse Mercator (UTM)**

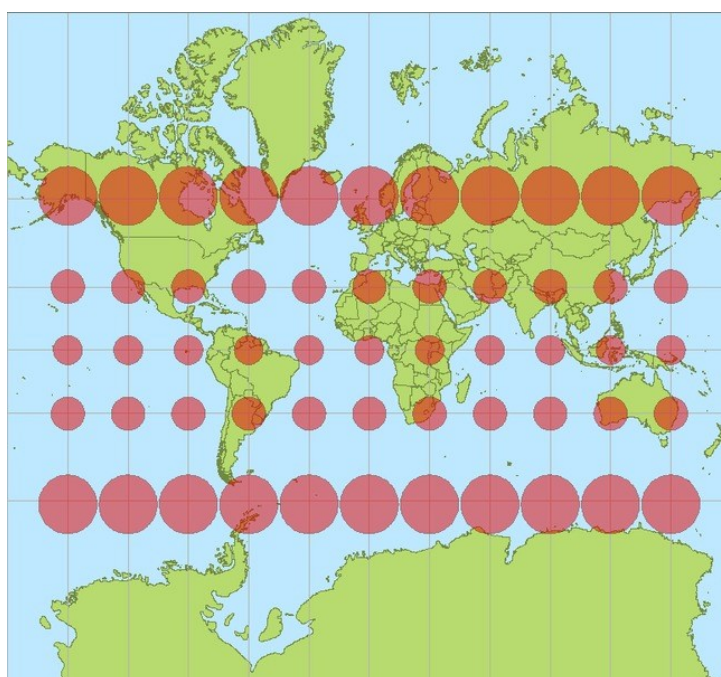
V rámci NATO se pro vojenské účely používá právě WGS-84 UTM. Jedná se o Mercatorovo válcové zobrazení v příčné poloze nad elipsoidem WGS-84. WGS-84 je jeden z nejčastěji používaných elipsoidů (EPSG 4326) v geoinformatické sféře. Souřadnice naměřené s pomocí GPS se vztahují právě k tomuto elipsoidu. Pokud jsou ve sféře GIS distribuována data v zeměpisných souřadnicích, souřadnicovým systémem je většinou právě WGS-84. Pod EPSG kódem 4979 je také WGS-84, oproti EPSG uvedenému v předchozím případě jsou však souřadnice 3D (použito v CesiumJS, navíc je ještě uvedena elipsoidální výška). Mercatorovo válcové zobrazení je konformní – zachovává úhly. WGS-84 UTM je jeden z nejpoužívanějších souřadnicových systémů. Celý svět je rozdělen na šestistupňové poledníkové pásy zobrazené na samostatné válcové plochy. Souřadnicový systém není vhodný pro mapy světa, používá se pro podrobné mapy velkých měřítek. Většina území ČR se nachází v zóně 33 N (EPSG 32633). [16]

### **2.6.4 Web Mercator**

Je souřadnicový systém známý také pod názvy Pseudo Mercator, Spherical Mercator nebo Google Mercator (EPSG 3857). V současnosti se jedná o nejpoužívanější souřadnicový systém pro online mapy. Příkladem mohou být Google Maps, Bing Maps, Mapy.cz, OpenStreetMap, podklady Esri aj. Jedná se v podstatě o standard pro



webovou kartografií. Taktéž ČÚZK poskytuje některé své produkty kromě S-JTSK také ve Web Mercatorovi. Souřadnicový systém se rozšířil společně s nástupem Google Maps v roce 2005. Použito je opět Mercatorovo zobrazení, nicméně souřadnice z elipsoidu WGS-84 se převádějí do roviny pomocí zobrazovacích rovnic pro kouli. Mercatorovo zobrazení je nevhodné používat ve vyšších zeměpisných šířkách, neboť je zde extrémní zkreslení (póly se zobrazují v nekonečnu). Toto má negativní důsledek na vizuální vjem mapy (Obrázek 11), jelikož např. Grónsko se jeví větší než Afrika. [1, 17]



**Obrázek 11: Tissotova indikatrix – Mercatorovo zobrazení [17]**

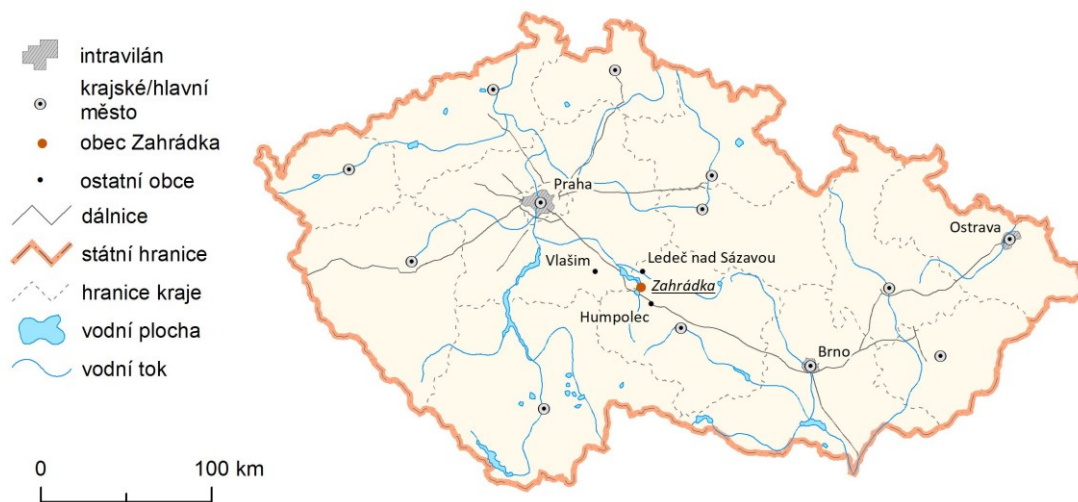
### 3 Metodika zpracování

Pro implementaci připadaly v úvahu dvě knihovny – CesiumJS a deck.gl. Nakonec bylo rozhodnuto využít právě CesiumJS z důvodu provázanosti s platformou Cesium Ion. Autor zároveň již měl zkušenosti se zmíněným řešením. Pro základní použití je cloud k dispozici zdarma. Z důvodu využití Cesium Ion, kde je uložen samotný 3D model, nemá aplikace žádný back end. Taktéž nebyl použit jakýkoliv front end framework (Bootstrap, jQuery aj.). Využit byl pouze lightbox2 pro obrázkovou galerii. Celá aplikace byla napsána v čistém JavaScriptu jako single-page aplikace.

#### 3.1 Charakteristika zájmového území

Městys Zahrádka se nacházel na Českomoravské vrchovině poblíž Ledče nad Sázavou (dnes Kraj Vysočina, okres Havlíčkův Brod, Obrázek 12). Obec byla dobře vybavena nejrůznějšími službami, jednalo se o spádové sídlo. K dispozici bylo i autobusové spojení do Prahy. V době před zbořením žilo v obci přes 1000 obyvatel. Městyssem protékala řeka Želivka, která měla vodu s vysokou jakostí. Právě z tohoto důvodu bylo rozhodnuto o výstavbě v. n. Švihov. Tato přehradní nádrž v současnosti zásobuje pitnou vodou cca 1,5 milionu lidí. V důsledku vytyčení ochranného pásma byla obec zbořena v podstatě celá, byť zatopená je jen menší část původního území. Na bývalém náměstí zůstal zachován kostel sv. Víta, socha Panny Marie, kašna a pomník padlým (později přesunut na hřbitov). Zmíněné objekty se dnes nachází jen pár desítek metrů od hladiny přehrady. Místo je možné navštívit. V průběhu roku pořádá Spolek Přátelé Zahrádky nejrůznější kulturní akce, kdy je otevřen i kostel.

[1] Autor se dlouhodobě zabývá zmíněným zájmovým územím. Více informací o místě je k dispozici na adrese [www.zahradkauledce.cz](http://www.zahradkauledce.cz).



Obrázek 12: Poloha obce Zahrádka [1]

### 3.2 Příprava dat

Tvorba samotného 3D modelu byla započata v rámci staršího autorova projektu. [1] Většina budov však byla vymodelována ve volném čase autora. Nejprve bylo třeba zgeoreferencovat příslušné podklady. Konkrétně se jednalo o Státní mapu odvozenou 1 : 5000 (SMO-5), Mapu evidence nemovitostí 1 : 2880, Vojenské Topografické mapy S-52 1 : 10000 a archivní letecké měřické snímky (rok 1966 a 1971), které spravuje VGHMÚř v Dobrušce. Archivní mapové podklady poskytuje ČÚZK. Za cílový souřadnicový systém byl zvolen WGS 84 UTM zone 33 N (EPSG 32633). [1]

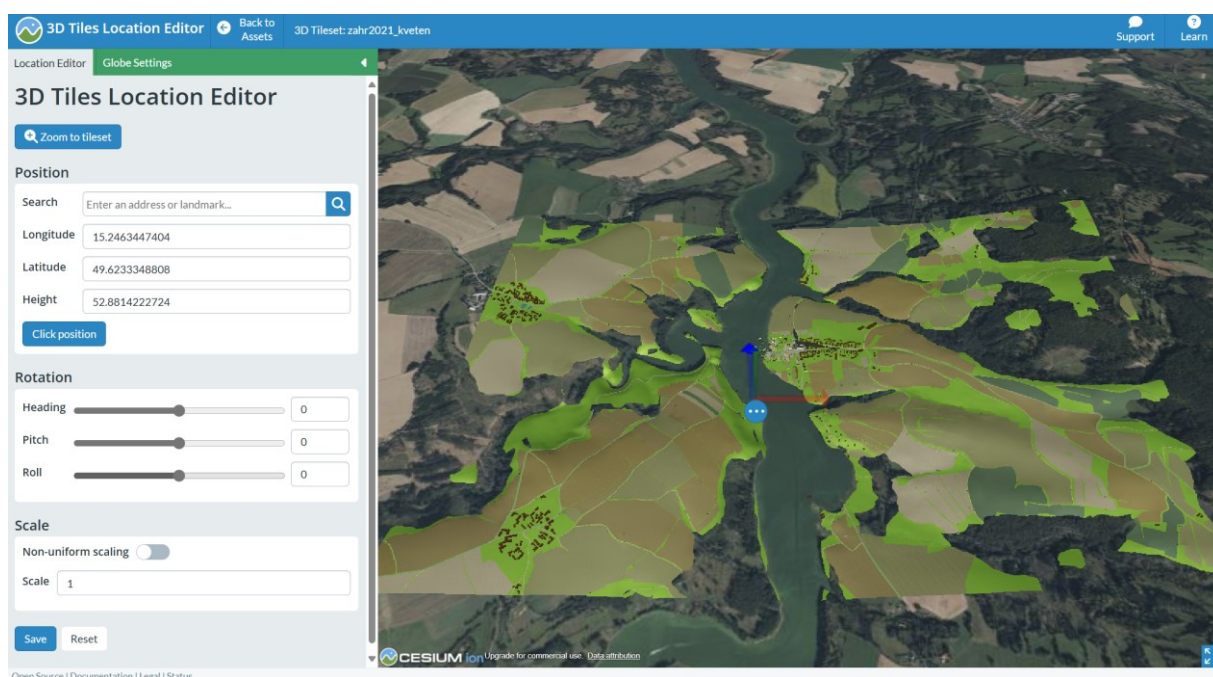
Pro zjištění rozměrů půdorysů budov a dalších objektů byly stěžejní právě zmíněné mapové podklady (SMO-5 a Katastrální mapa). Bohužel množství objektů nebylo zkolaudováno, a proto ve zmíněných mapových dílech nebyly domy zachyceny. V tomto směru byly velmi užitečné letecké měřické snímky. Pro zjištění výškových poměrů v zatopeném území byly zvektorizovány vrstevnice z topografických map S-52. V nezatopeném území byla použita data laserového skenování terénu (DMR 5G). Jedná se o produkt poskytovaný ČÚZK. Výškopis je reprezentován velmi hustou

nepravidelnou sítí výškových bodů. Souřadnicovým systémem je S-JTSK, výškovým systémem Balt po vyrovnání. Po transformaci dat do WGS-84 UTM zone 33 N byl ze zmíněných podkladů interpolován TIN. Použity byly také automatizovaně generované povinné hrany (breaklines vytvořené pomocí Python skriptů) pro modelování silničních těles a koryt vodních toků. Pro TIN interpolaci byl využit program QGIS.

3D model byl vytvořen v SW Blender, použity byly doplňky jako například Blender GIS (import GIS dat do Blenderu) či fSpy, který slouží pro fotogrammetrickou modelaci objektů z jediného snímku. Toto modelování využívá perspektivní zkreslení objektů na snímku. Socha Panny Marie byla vytvořena metodou Structure from Motion v programu Meshroom. [1] Vegetace byla do modelu krajiny vložena v podobě billboard modelů pro co možná nejlepší datovou optimalizaci. Tvorba 3D modelu celkově zabrala stovky hodin času. Podrobně je přesto vymodelována pouze menšina budov hlavně na náměstí a v okolí. Podkladové archivní snímky pro modelaci laskavě poskytl Spolek Přátelé Zahrádky.

Výsledný 3D model byl vyexportován z Blenderu do binárního glTF s texturami v separátním adresáři (glTF Separate). Takto byl model nahrán na Cesium Ion. Následně byl zgeoreferencován s pomocí interaktivního georeferenceru (Adjust Tileset Location) přímo v prostředí Cesium Ion (Obrázek 13). Nakonec byl ještě vygenerován přístupový token (Access Tokens → Create tokens) pro umožnění načítání modelu do vlastní aplikace. Vegetace byla oddělena do separátního modelu. Toto umožnilo vytvořit vrstvu, kterou je možno zcela vypnout a dosáhnout tak nižší náročnosti na výpočetní výkon a zároveň lze lépe prohlížet budovy jinak zastíněné stromy. 3D model představující stav území v 50. letech 20. stol. byl vytvořen pouze jako DMT s archivním ortofotem. Ortofoto vytvořila agentura Cenia a je k dispozici zdarma pod svobodnou licencí v S-JTSK. Obdobně vznikl 3D model reprezentující současný stav. Aktuální ortofoto je k dispozici ke stažení na stránkách ČÚZK. Obě ortofota musela být před použitím transformována do EPSG 32633 v SW QGIS.

V neposlední řadě byly k dispozici body zájmu ve formátu GeoJSON v zeměpisných souřadnicích (WGS-84, EPSG 4326). Každý bod měl atributy vztahující se k jeho nadmořské výšce, názvu místa, popisu místa a stavu (zaniklý/dochovaný/nově postavený objekt). Tato bodová data a kartografické značky byly využity již v interaktivní mapě Zahrádky. [1] Pro účely této práce musely být bodům přiřazeny nadmořské výšky z DMT s pomocí funkce Drape v programu QGIS. Tyto výšky byly ještě zvětšeny, aby se body nacházely nad úrovní terénu.



**Obrázek 13: Georeferencování 3D modelu v Cesium Ion, vlastní tvorba**

### **3.3 Implementace v CesiumJS**

#### **3.3.1 Inicializace 3D okna**

Pro zobrazení interaktivního okna s 3D terénem a mapovými podklady je nutné v těle stránky vytvořit příslušný `DIV` oddíl. V samotném skriptu je třeba specifikovat přístupové tokeny k jednotlivým službám (např. Cesium Ion – `Cesium.Ion.defaultAccessToken`). Následně je možné vytvořit instanci třídy `Viewer`. Tento krok vede k zobrazení interaktivního okna na stránce. Přímo v konstruktoru



je specifikováno id `DIV` oddílu, kde se má okno vykreslit a objekt s dalším nastavením `Vieweru`. Jedná se například o atributy `terrainProvider` (objekt, načítaný terén), `terrainShadows` (zapnutí stínů terénu) nebo boolean atributy, kde je možno zapnout/vypnout jednotlivé prvky uživatelského rozhraní – `baseLayerPicker`, `sceneModePicker` a další.

Pro lepší vizuální zážitek byly zapnuty některé efekty, jejichž ovládání je dostupné přes atributy `Vieweru` a `Scene`. Jedná se např. o `viewer.shadows` (Obrázek 14), `scene.fog`, `scene.skyAtmosphere`, `scene.highDynamicRange` aj. Taktéž byl nastaven čas ve scéně na 12 hodin pro co nejlepší osvětlení 3D modelu. Toho bylo dosaženo přes atribut `currentTime` instance třídy `Clock`. Čas je jinak samozřejmě možné měnit ručně na časové ose dole v aplikaci. Do aplikace bylo přejato ovládání s pomocí kláves WSAD z tutoriálové aplikace dostupné na Cesium SandCastle. [18]



**Obrázek 14: Náměstí se stíny/bez stínů, vlastní tvorba**

### 3.3.2 Přidání podkladů

Body zájmu byly načteny do aplikace asynchronně z GeoJSON souboru. Kód tak musel být ošetřen s pomocí `try/catch`. Do scény byly body přidány metodou `add(pointsGeoJson)`. Následně bylo skrz body iterováno ve `for` cyklu. Každému bodu ještě musela být přiřazena správná nadmořská výška, jelikož Cesium používá jiný výškový systém. Následovalo nastavení obsahu pop up okna z atributů bodů a nastavení příslušné kartografické značky. Taktéž bylo zapnuto clusterování bodů

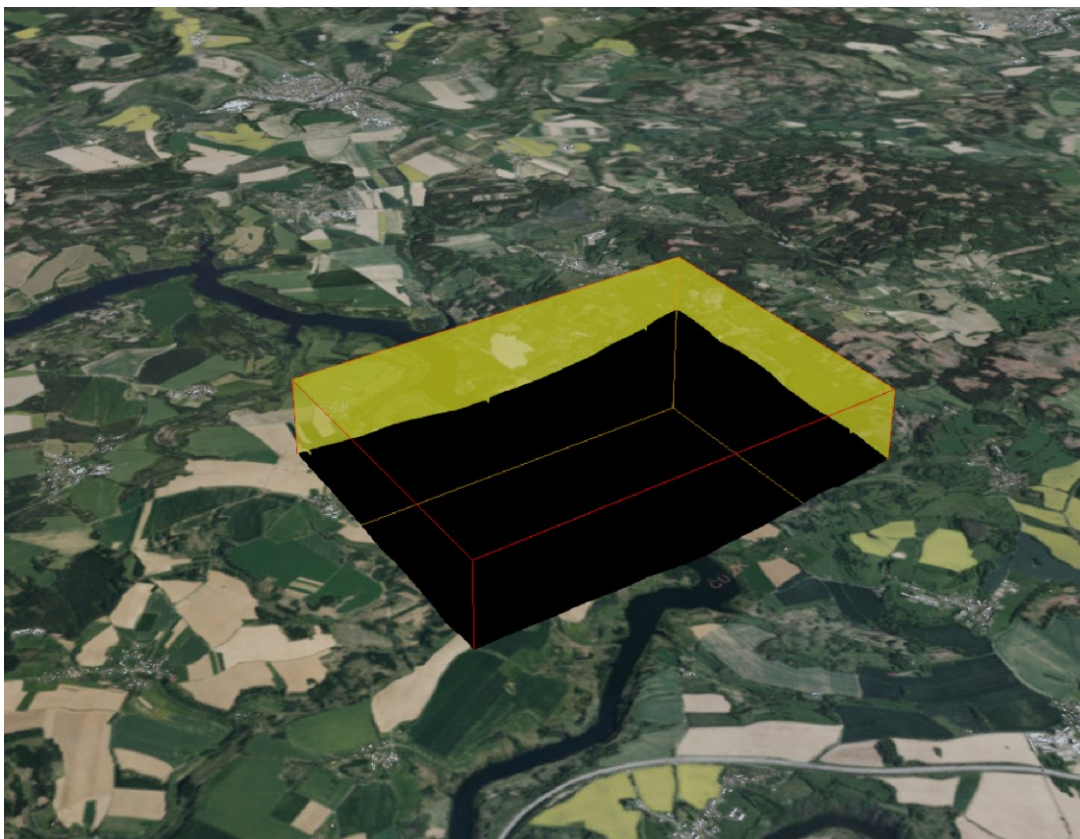
(`pointsGeoJson.clustering.enabled`). Kartografické značky měly být původně uvnitř špendlíků tvořených s pomocí třídy `PinBuilder`. Bohužel se vykreslovaly pouze jako bílá plocha uvnitř špendlíku (Obrázek 15). Proto bylo od tohoto řešení upuštěno.



**Obrázek 15: PinBuilder, vlastní tvorba**

Následně byly do scény opět asynchronně přidány jednotlivé 3D modely. Jedná se o 3D model území v 50. letech, 60. letech, v současnosti, a ještě navíc samostatně model vegetace pro umožnění vypínání stromů ve scéně. Modely byly přidány jakožto instance `Cesium3DTileset`. Tato třída má metodu pro načtení modelu z Cesium Ion `fromIonAssetId()`, jejímž argumentem je ID příslušného modelu ve službě Ion.

Jelikož DMT byl po ručních úpravách v Blenderu pouze obecný mesh, nikoliv TIN či raster, byl DMT přímo součástí daných 3D modelů. Pro korektní zobrazení 3D modelu s DMT musel být v DMT poskytovaném Cesium vytvořen otvor (Obrázek 16), aby bylo zamezeno překrývání poskytovaného a vlastního DMT. Otvor byl vytvořen s pomocí `ClippingPlane`. Poskytovaný DMT je jinak zobrazen v okolním území mimo Zahrádku.



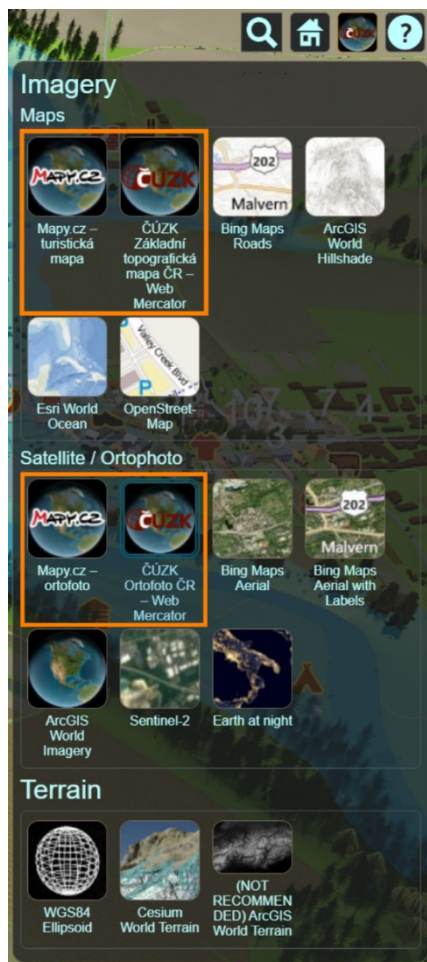
**Obrázek 16: Vizualizace otvoru v Cesium DMT pro 3D model Zahrádky, vlastní tvorba**

Vrstva přehrady znázorněná průhlednou vrstvou byla do scény přidána jakožto instance třídy `Entity`. Entitě se dá přes atributy nastavit název, popis, id, grafická reprezentace (geometrický útvar), její poloha aj. Informace (název, popis) se zobrazí v pop upu po kliknutí na přehradu. Výběr entity bohužel nejde zakázat. V praxi je následkem otevření pop upu přehrady, pokud uživatel klikne na bodovou značku umístěnou pod průhlednou hladinou přehrady. Problém byl vyřešen zvýšením nadmořské výšky bodů zájmu tak, aby byly všechny nad úrovní hladiny.

V neposlední řadě byly do aplikace přidány mapové podklady a ortofota ČÚZK a map.cz. ČÚZK poskytuje tyto vrstvy mimo jiné jako WMTS služby v souřadnicovém systému Web Mercator. Pro WMTS byla využita třída `WebMapTileServiceImageryProvider`, přičemž v konstruktoru této třídy je nastavení daných služeb (URL apod.). Pro zobrazení služeb v nabídce vrstev je nutné vytvořit `ProviderViewModel` a ten následně přidat do pole v atributu instance třídy `BaseLayerPickerViewModel`. Takto se uživateli zobrazí přidávané vrstvy přímo



v BaseLayerPickeru (Obrázek 17), který je defaultně součástí rozhraní CesiumJS. Obdobně byly přidány podklady map.cz, pouze místo třídy `WebMapTileServiceImageryProvider` byla použita třída `UrlTemplateImageryProvider`, jelikož v případě map.cz se nejedná o WMTS službu.



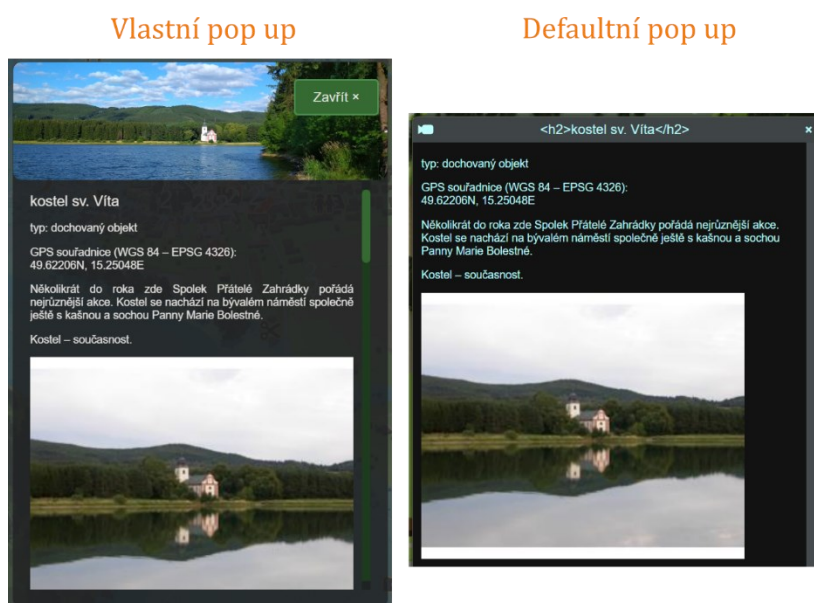
Obrázek 17: Výběr podkladů ČÚZK a mapy.cz, vlastní tvorba

### 3.3.3 Vlastní uživatelské rozhraní a obsluha událostí

Pro tvorbu vlastního pop up okna byl vytvořen oddíl `popupIntroduction`, který má fixní pozici a zaujímá celou stránku, přičemž standardně není zobrazován (`display: none;`). Tento oddíl je šedý průhledný a tvoří pozadí samotného pop upu. Právě v `popupIntroduction` je vložen oddíl `popupWindow`, což je samotné pop up okno. Pro zobrazení pop upu je nutné `popupIntroduction` přiřadit třídu `showPopup`

(`popupIntroduction.classList.add("showPopup")`) s jedinou vlastností `display: block;`. Pro zavření okna je třída naopak odebrána.

Pop up slouží pro zobrazení úvodních informací o aplikaci. Taktéž do vlastního pop upu byl přesunut popis ovládání, který je jinak standardně součástí nápovědy k CesiumJS aplikaci a zobrazí se po kliknutí na otazník v pravém horním rohu aplikace. Obsluha právě tohoto tlačítka byla změněna a tlačítko tak slouží pro zobrazení vlastního pop upu s informacemi a nápovědou. Nejdůležitější funkcí pop upu je však zobrazování informací o jednotlivých entitách (bodech zájmu). Pop up defaultně přístupný v CesiumJS je graficky strohý (Obrázek 18).



Obrázek 18: Vlastní vs Defaultní pop up, vlastní tvorba

U instance `viewer` byla změněna defaultní obsluha události po kliknutí na bod zájmu tak, aby byl otevřen vlastní pop up. Taktéž obsluha tlačítka „Domů“ byla změněna, aby po kliknutí na něj byl přenastaven pohled uživatele na Zahrádku s okolím. V defaultním nastavením je uživatel přesměrován na Severní Ameriku.

Pro ovládání scény byl vytvořen vlastní panel s nastavením umístěný vlevo na stránce. Pro volbu časového období modelu byl použit `radio button`. Při volbě konkrétního období je u `Cesium3DTiles` změněna hodnota v atributu `show`

(`true/false`). Obdobně bylo použito množství `checkbox`ů pro zapínání/vypínání jednotlivých prvků scény (stíny, OpenStreetMap budovy, v. n. Švihov, body zájmu, terén aj.). Pro změnu nadmořské výšky hladiny Švihova byl využit `slider`. U všech `slider`ů byly obslouženy události pro „`mousemove`“ i „`click`“. U entity přehrady je pak měněna výška vytvořením nové instance `Cartesian3` (3D poloha) s jinou nadmořskou výškou. Tato 3D poloha je uložena v atributu entity `position`. Pro změnu průhlednosti s pomocí `slideru` byla vytvořena nová barva z původní barvy úpravou alfa kanálu metodou `withAlpha()`. Nastavení rozlišení scény se provádí s pomocí úpravy hodnoty atributu `resolutionScale` instance `viewer`. Při nastavení maximální hodnoty je scéna renderována v nativním rozlišení monitoru. Nakonec byla vytvořena nová instance třídy `Credit("© Univerzita Hradec Králové", true)`. Tato instance byla do scény přidána s pomocí metody `addStaticCredit()` třídy `CreditDisplay`, která sdružuje afiliace jednotlivých poskytovatelů dat a další informace. Ukázky zdrojového kódu jsou v příloze práce.

## 4 Testování a výsledky

### 4.1 Měření FPS

Aplikace byla testována s pomocí měření FPS na dvou PC. V prvním případě se jednalo o výkonnou sestavu s moderním HW, druhým testovacím strojem byl již zastaralý notebook (Tabulka 1). Knihovna CesiumJS disponuje zabudovaným FPS (Frames per second) měřičem. Při statické scéně bylo na PC 1 stabilních 60 FPS, při pohybu se FPS držely nad 50, pouze výjimečně spadly na 45 v nativním 4K rozlišení. PC 2 bylo značně problematictější. Při pohybu ve scéně při nativním Full HD rozlišení nebylo dosaženo více než 10 FPS. Při snížení rozlišení v CesiumJS na minimum se hodnoty zvýšily na 20 FPS, což již činilo aplikaci použitelnou pro běžného uživatele. Aplikace je tak poměrně náročná na grafický výkon zařízení. Při běhu na méně výkonném stroji je proto doporučeno snížit rozlišení scény a případně vypnout méně důležité prvky. Po vypnutí vegetace, anti-aliasingu, stínů a mlhy na obzoru se podařilo dosáhnout na druhém testovaném PC nad 40 FPS, jednalo se v podstatě o plynulý průchod scénou. Specifikace daných PC jsou uvedeny níže (Tabulka 1).

**Tabulka 1: Specifikace PC**

	<i>PC 1 (desktop)</i>	<i>PC 2 (laptop)</i>
<i>CPU</i>	AMD Ryzen 7900	Intel Core i7 6500U
<i>GPU</i>	RTX 3080 10 GB	GTX 950M 4 GB
<i>MB</i>	Asus TUF Gaming X670e	HP OEM
<i>RAM</i>	32 GB DDR5	16 GB DDR3
<i>PSU</i>	850 W	adaptér 90 W
<i>OS</i>	Win. 11 Pro	Win. 10 Home
<i>Rozlišení monitoru</i>	4K	Full HD

**Zdroj: vlastní tvorba**

## 4.2 Výsledná implementace a 3D vizualizace

Výstupem je funkční implementace webové interaktivní 3D aplikace zobrazující prostorový model zaniklé obce Zahrádka u Ledče nad Sázavou vytvořená s pomocí knihovny CesiumJS. Aplikace je k dispozici na adrese [www.rostislavberka.cz/3d.html](http://www.rostislavberka.cz/3d.html) (Obrázek 19).

**Popis aplikace a legenda (klikněte pro zvětšení)**

**vyhledání adresy**  
**výchozí pozice kamery**  
**nápověda**  
**mapové podklady**  
**na celou obrazovku**  
**VR**  
**nastavení konkrétního data a hodiny scény**  
**zapnutí animace scény**  
**rychlost animace**

**Legendu:** (objekty rozlišeny barvou ohraničení)    dochované objekty    zaniklé objekty    nové objekty

**Ovládání klávesnicí - rozšiřující k myši**

**Q** **W** **E**  
**A** **S** **D**

Q,E: nahoru, dolu    Šipky (alternativně W,S,A,D): dopředu, dozadu, doprava, doleva

V okolním terénu lze přepínat mapové podklady - tlačítko vpravo nahoře  
Vyzkoušejte Mapy.cz nebo mapy ČÚZK

sořadnicový systém aplikace:  
WGS 84 Web Mercator

Univerzita Hradec Králové  
Fakulta informatiky a managementu

Obrázek 19: Popis aplikace, vlastní tvorba

I přes bezesporu efektivní vizualizaci území trpí aplikace a 3D model některými problémy (Obrázek 20). Například v DMT v CesiumJS vznikaly nejrůznější štěpy a jednotlivé povrchy evidentně nelicují. Taktéž s texturami byl problém a v některých místech docházelo k jejich nekorektnímu vykreslování. Tyto artefakty jsou



pravděpodobně důsledkem optimalizace výkonu aplikace. Taktéž vlastní DMT a DMT poskytovaný CesiumJS vzájemně nenavazují. DMT používají různá podkladová data, přičemž vlastní DMT by měl být výrazně přesnější a detailnější. Z tohoto důvodu nebyly v zájmovém území vizualizovány budovy OpenStreetMap, jelikož jejich nadmořské výšky jsou převzaty z DMT dostupného v CesiumJS.



**Obrázek 20: Blender vs CesiumJS, vlastní tvorba**

Pro vizualizaci současného stavu území byl použit vlastní DMT vytvořený z dat DMR 5G. Terén dostupný na Cesium Ion je velmi nepřesný a nepříliš kvalitní. Nejedná se pouze o drobný estetický problém, u zmíněného terénu je problematické

i zachycení elementárních topografických prvků krajiny jako jsou například údolí vodních toků nebo přehradní údolí (Obrázek 21).



**Obrázek 21: Nekvalitní DMT poskytovaný Cesium, vlastní tvorba**

Samotná knihovna CesiumJS není responzivní. Veškeré ovládací prvky defaultně přítomné v GUI při zobrazení 3D scény jsou velmi malé na mobilních telefonech. Z tohoto důvodu nebyla ani vlastní aplikace přizpůsobována mobilním zařízením.

Aplikace je jinak poměrně náročná na výpočetní výkon PC, a proto je lepší ji spouštět na počítači s dedikovanou grafickou kartou.

Aplikace vizualizuje skutečně detailní a rozměrný model skládající se z desítek tisíc vertexů a trojúhelníků (Tabulka 2). Je zřejmé, že vypnutí vegetace výrazně sníží náročnost scény.

**Tabulka 2: Počty geometrických primitiv modelu**

	<i>DMT s budovami</i>	<i>Vegetace</i>
<i>Vertexy [tis.]</i>	151	405
<i>Hrany [tis.]</i>	278	432
<i>Trojúhelníky [tis.]</i>	289	224

**Zdroj: vlastní tvorba**

Některé objekty v 3D modelu se také zobrazují nekorektně v nepřírozeně tmavých odstínech (neodráží světlo). Tento problém se vyskytl pouze u jednotek budov (Obrázek 22). Příčinu se nepodařilo zjistit.





**Obrázek 22: Budovy neodrážející světlo (vpravo), vlastní tvorba**

## 5 Závěr

V práci byly prozkoumány možnosti vizualizace rozsáhlých detailních georeferencovaných 3D modelů na webu. V teoretické části práce jsou rozebrány specifika dané problematiky jako například 3D Tiles, používané datové formáty, knihovny či souřadnicové systémy. Praktická implementace zahrnuje aplikaci naprogramovanou v JavaScriptu s pomocí knihovny CesiumJS. Díky využití cloudu Cesium Ion nebylo třeba programovat back end. Právě existence služby Ion byla jedním z důvodů pro využití knihovny CesiumJS. Na front endu nebyly použity žádné frameworky. Aplikace umožňuje uživateli prohlížet 3D model zaniklé obce Zahrádka u Ledče. Ukázky kódu zahrnující nejdůležitější součásti webové 3D vizualizace jsou uvedeny v příloze práce.

Podkladovými daty byl v první řadě 3D model vytvořený v Blenderu. DMT byl interpolován z dat leteckého laserového skenování a vrstevnic starých map. Samotné budovy byly vytvořeny kombinací fotogrammetrických metod a ručního modelování. Z dalších dat lze uvést body zájmu ve formátu GeoJSON v souřadnicovém systému WGS-84 (zeměpisné souřadnice).

Aplikace byla otestována na dvou PC. V případě výkonného zařízení se jednalo o plynulý uživatelský zážitek při pohybu scénou. Problematictější byl starší notebook. Pro plynulé prohlížení 3D modelu bylo třeba snížit náročnost scény (snížit rozlišení, vypnout některé prvky aj.). Z tohoto důvodu by bylo vhodné zkoumat další možnosti optimalizace scény.

Aplikace je plně funkční. Z hlediska budoucího rozvoje je záměrem dokončit prezentovaný 3D model. Většina budov není do detailu vymodelována a je nahrazena pouze půdorysnými modely. Podrobně bylo zhotoveno náměstí, které bylo centrem obce. Z hlediska funkcionality splnila aplikace vše, co od ní autor očekával.

## 6 Seznam použité literatury

- [1] BERKA, Rostislav. Využití soudobých 3D geoinformačních technologií pro historické rekonstrukce území: případová studie Zahrádka u Ledče nad Sázavou. Praha, 2020. Bakalářská práce. Univerzita Karlova.
- [2] KŘÍŽEK, Marek, Tomáš UXA a Peter MIDA. Praktikum morfometrických analýz reliéfu. Praha: Nakladatelství Karolinum, 2016.
- [3] PACINA, Jan a Marcel BREJCHA. Digitální modely terénu. Ústí nad Labem: Univerzita Jana Evangelisty Purkyně, 2014.
- [4] WIKIPEDIA. Delaunay triangulation [online]. [vid. 2024-06-10]. Dostupné z: [https://en.wikipedia.org/wiki/Delaunay\\_triangulation](https://en.wikipedia.org/wiki/Delaunay_triangulation)
- [5] HOETMER, Ken. Announcing deprecation of the Google Earth API [online]. [vid. 2024-06-14]. Dostupné z: <https://mapsplatform.googleblog.com/2014/12/announcing-deprecation-of-google-earth.html>
- [6] KRYSHENYK, Pavlo. 3D VIZUALIZACE REKONSTRUOVANÝCH SÍDEL VE WEBOVÉM PROSTŘEDÍ. Praha, 2013. Bakalářská práce. Univerzita Karlova.
- [7] CESIUM. The Platfotm for 3D Geospatial [online]. [vid. 2024-06-14]. Dostupné z: <https://cesium.com/>
- [8] NEKOLA, Lukáš. NÁVRH APLIKACE PRO VIZUALIZACI GEOGRAFICKÝCH DAT DOKUMENTUJÍCÍCH VÝVOJ ZANIKLÝCH KRAJIN. Praha, 2023. Diplomová práce. Univerzita Karlova.
- [9] BRŮHA, Lukáš. Zaniklé krajiny Česka (interaktivní mapová aplikace) [online]. [vid. 2024-06-14]. Dostupné z: <http://www.zaniklekrajiny.cz/vystupy/interaktivni-mapova-aplikace>
- [10] DECK.GL. deck.gl is a GPU-powered framework for visual exploratory data analysis of large datasets. [online]. [vid. 2024-06-14]. Dostupné z: <https://deck.gl/>
- [11] OPEN GEOSPATIAL CONSORTIUM. Open Geospatial Consortium [online]. [vid. 2024-06-18]. Dostupné z: <https://www.ogc.org/>
- [12] SAVOLAINEN, Sampo. Monitoring of ArcGIS WMTS services [online]. [vid. 2020-03-19]. Dostupné z: <https://www.spatineo.com/monitoring-arcgis-wmts/>
- [13] ESRI. What is KML? [online]. [vid. 2024-06-23]. Dostupné z: <https://pro.arcgis.com/en/pro-app/latest/help/data/kml/what-is-kml-.htm>
- [14] KHROSOS GROUP. Khronos Group [online]. [vid. 2024-06-23]. Dostupné z: <https://www.khronos.org/>

- [15] ADOBE. Soubory Collada [online]. [vid. 2024-06-23]. Dostupné z: <https://www.adobe.com/cz/creativecloud/file-types/image/vector/collada-file.html>
- [16] MIKLÍN, Jan, Radek DUŠEK, Luděk KRTIČKA a Oto KALÁB. Tvorba map. 1. vydání. Ostrava: Ostravská univerzita, 2018. ISBN 978-80-7599-017-4.
- [17] NÉTEK, Rostislav. Webová kartografie – specifika tvorby interaktivních map na webu. 1. vydání. Olomouc: Univerzita Palackého v Olomouci, 2021. ISBN 978-80-244-5827-4.
- [18] CESIUM. Cesium SandCastle [online]. [vid. 2024-07-04]. Dostupné z: <https://sandcastle.cesium.com/?src=Camera%20Tutorial.html>

## 7 Přílohy

Ukázky zdrojového kódu aplikace:

- 1) Inicializace 3D okna
- 2) Přidání podkladů
- 3) Vlastní uživatelské rozhraní a obsluha událostí

Aplikace je k dispozici na adrese: [www.rostislavberka.cz/3d.html](http://www.rostislavberka.cz/3d.html) Kompletní zdrojový kód je v externí příloze práce.

## Ukázky zdrojového kódu aplikace

Jednotlivé ukázky odpovídají popisu v metodické části práce, kde jsou okomentovány.

### 1) Inicializace 3D okna

Vytvoření instance terénu a interaktivního 3D okna:

```
const terrainProvider = await
Cesium.CesiumTerrainProvider.fromIonAssetId(1, {
  requestVertexNormals: true, // required for Lighting
  requestWaterMask: true, // better water effects (just seas)
});
// "cesiumContainer" - ID of HTML container
const viewer = new Cesium.Viewer("cesiumContainer", {
  vrButton: true,
  terrainProvider: terrainProvider,
  terrainShadows: 1,
  scene3DOnly: true,
  sceneModePicker: false,
  baseLayerPicker: true,
  requestRenderMode: true,
  maximumRenderTimeChange: Infinity,
  msaaSamples: 1, // better low for better performance
  imageryProvider: true, // false for testing purposes
});
```

Zapnutí nejrůznějších efektů:

```
viewer.shadows = true;
viewer.shadowMap.softShadows = false; // doesnt look good
viewer.shadowMap.darkness = 0.5;
viewer.scene.fog.enabled = true;
viewer.scene.highDynamicRange = true; // HDR
viewer.scene.skyAtmosphere.show = true;
```

```
viewer.scene.postProcessStages.fxaa.enabled = true;
```

```
const globe = viewer.scene.globe;
globe.enableLighting = true;
// demanding for computer:
globe.showGroundAtmosphere = true;
globe.dynamicAtmosphereLighting = true;
globe.dynamicAtmosphereLightingFromSun = true;
```

Nastavení času ve scéně:

```
const currentDate = new Date();
currentDate.setHours(12); // at 12 o'clock is best lightning for 3D
model
currentDate.setMinutes(0);
currentDate.setSeconds(0);
viewer.clock.currentTime= new Cesium.JulianDate.fromDate(currentDate);
```

## 2) Přidání podkladů

Přidání bodů zájmu:

```
try { // try/catch and global var --> because GeoJson is loaded as
promise
  var pointsGeoJson = await
Cesium.GeoJsonDataSource.load(interestPointsPath);
  viewer.dataSources.add(pointsGeoJson);
  const pointsEntities = pointsGeoJson.entities.values;
  for (let i = 0; i < pointsEntities.length; i++) {
    let entity = pointsEntities[i];
    // set position (Z coordinate)
    const cartesianCoords = new
Cesium.Cartesian3(entity.position._value.x, entity.position._value.y,
entity.position._value.z);
    const cartographicCoords =
Cesium.Ellipsoid.WGS84.cartesianToCartographic(cartesianCoords);
    const longitude =
Cesium.Math.toDegrees(cartographicCoords.longitude);
    const latitude =
Cesium.Math.toDegrees(cartographicCoords.latitude);
    entity.position = Cesium.Cartesian3.fromDegrees(longitude,
latitude,
entity.properties.Z_modified + elevationDiff);
```



```

    // set info in pop up
    entity.name = "<h2>" + entity.properties.nazev_sluzby + "</h2>";
    // show attributes of geojson in pop up
    const type = "<p>typ: " + entity.properties.stav; + "</p>";
    const coords = "<p>GPS souřadnice (WGS 84 - EPSG 4326): <br>" +
    lat + "N, " + lon + "E </p>";
    const info = "<p>" + entity.properties.popupContent + "</p>";
    entity.description = type + coords + info;
    // set marker
    let markerPath = markersPath + entity.properties.sluzba + ".svg";
    markerPath = markerPath.replace(/[0-9]/g, '');

    let multiply = 1;
    if (mobileDevice) {multiply = 1.75;}
    let scale;
    if (entity.properties.nazev_sluzby == "kostel sv. Víta") {
        scale = 0.35 * multiply; // marker of church is bigger to be
    dominant
    } else {
        scale = 0.15 * multiply;
    }
    entity.billboard = {
        image: markerPath,
        sizeInMeters: false, // great feature!!! can relative / absolute
        size of icon, width and height have to be but not suitable for my app
        //width: 20,
        //height: 20,
        scale: scale// 0 - 1.0 --> smaller img, 1< --> bigger img
    }
}
pointsGeoJson.clustering.enabled = true;
pointsGeoJson.clustering.pixelRange = 0.5;
pointsGeoJson.clustering.minimumClusterSize = 2;
}
catch (error) {
    console.log(error);
}
}

```

Přidání 3D modelu Zahrádky v 60. letech 20. stol.:

```

const zahrHistoricalTileset = await
Cesium.Cesium3DTileset.fromIonAssetId(zahradkaAssetId, {
    showCreditsOnScreen: false,
    preloadWhenHidden: true,
    preloadFlightDestinations: true
});
viewer.scene.primitives.add(zahrHistoricalTileset);

```

## Vytvoření otvoru v Cesium DMT:

```

const bbox = new Cesium.Entity({ // definition of o box
  position: zahradkaCentroid,
  show: false,
  box: {
    dimensions: new Cesium.Cartesian3(1000, 1000, 1500), // size of
    box
    material: Cesium.Color.YELLOW.withAlpha(0.3),
    outline: true,
    outlineColor: Cesium.Color.RED,
  },
});
viewer.entities.add(bbox);
function createClippingPlanes(halfX, halfY) {
  return new Cesium.ClippingPlaneCollection({
    modelMatrix: bbox.computeModelMatrix(Cesium.JulianDate.now()),
    planes: [
      new Cesium.ClippingPlane(new Cesium.Cartesian3(1.0, 0.0, 0.0), -
halfX),
      new Cesium.ClippingPlane(new Cesium.Cartesian3(-1.0, 0.0, 0.0),
-halfX),
      new Cesium.ClippingPlane(new Cesium.Cartesian3(0.0, 1.0, 0.0), -
halfY),
      new Cesium.ClippingPlane(new Cesium.Cartesian3(0.0, -1.0, 0.0),
-halfY),
    ],
    edgeWidth: false ? 1.0 : 0.0,
    edgeColor: Cesium.Color.RED,
    enabled: true,
  });
}
const DTMclippingPlanes = createClippingPlanes(1800, 1280);
const OSMclippingPlanes = createClippingPlanes(1842, 1323);
globe.clippingPlanes = DTMclippingPlanes;
OSMbuildingTileset.clippingPlanes = OSMclippingPlanes;

```

## Přidání vrstvy přehrady:

```

const zahradkaCentroid = Cesium.Cartesian3.fromDegrees(lon, lat,
elevation); // mid point of Zahradka 3D model
const damColor = new Cesium.Color(0, 0.75, 1, 0.4); // R, G, B, alpha

function createSvihovDam(centroid, color) {
  return new Cesium.Entity({
    id: "svihovDam",
    name: "<br>v. n. Švihov",
    description: "<p>Klikli jste na hladinu v. n. Švihov. Pokud jste
chtěli kliknout na bodovou značku, která je pod hladinou, vypněte
vrstvu přehrady v levém ovládacím panelu.</p><p>Ve výchozím stavu je

```

```

vrstva přehrady v úrovni při běžném vodním stavu v. n. Švihov (376,8 m
n.m.).</p>",
  position: centroid,
  show: true,
  plane: {
    plane: new Cesium.Plane(Cesium.Cartesian3.UNIT_Z, 0.0),
    dimensions: new Cesium.Cartesian2(3600, 2652), // dimensions of
plane (X Y) (3600, 2652) for fitting Zahradka area
    fill: true,
    // material: Cesium.Color.DEEP_SKY_BLUE.withAlpha(0.2),
    material: color,
    outline: false,
    outlineColor: Cesium.Color.BLUE,
  },
});
let svihovDam = createSvihovDam(zahradkaCentroid, damColor);
viewer.entities.add(svihovDam);

```

### Přidání WMTS služby:

```

const ortofotoCUZKwmts = new Cesium.WebMapTileServiceImageryProvider({
  url:
"https://ags.cuzk.cz/arcgis1/rest/services/ORTOFOTO_WM/MapServer/WMTS"
,
  layer: "ORTOFOTO_WM",
  style: "default",
  format: "image/jpeg",
  tileMatrixSetID: "default028mm",
  maximumLevel: maximumLevel,
  credit: cuzkCredit
});
const ortofotoCUZKlayer = new Cesium.ProviderViewModel({
  name: "ČÚZK Ortofoto ČR - Web Mercator",
  iconUrl: cuzkLogo,
  category: "Satellite / Ortophoto",
  tooltip: "Prohlížečí služba WMTS - Ortofoto ČR - měřítková řada
Google Maps - Web Mercator, poskytováno ČÚZK, celá ČR pravidelně
snímána s periodicitou 2 roky, polohová přesnost cca 0,2 m.",
  creationFunction: () => {
    return ortofotoCUZKwmts;
  }
});
// layerObjects.push(ortofotoCUZKlayer);
layerObjects.unshift(ortofotoCUZKlayer);

```

### 3) Vlastní uživatelské rozhraní a obsluha událostí

Změna obsluhy události u tlačítka s otazníkem v GUI v CesiumJS:

```
viewer.navigationHelpButton.viewModel.command.beforeExecute.addEventListener(function (event) {
  event.cancel = true;
  popupText.innerHTML = infoText;
  popupIntroduction.classList.add("showPopup");
  addHelpEventListeners();
  popupText.scroll({
    top: 0
  });
});
```

Změna obsluhy tlačítka „Domů“:

```
const homePosition = {
  destination: Cesium.Cartesian3.fromDegrees(15.2491, 49.605, 1500),
  orientation: {
    heading: Cesium.Math.toRadians(0.0),
    pitch: Cesium.Math.toRadians(-30.0),
    roll: Cesium.Math.toRadians(0)
  },
  duration: 3 // how quick is the animation
}

viewer.homeButton.viewModel.command.beforeExecute.addEventListener(function (event) {
  event.cancel = true; // cancel all previous events
  viewer.scene.camera.flyTo(homePosition);
});
```

Obsluha slideru pro změnu nadmořské výšky hladiny Švihova:

```
let newCentroid;
let previousZ = elevation;

function changeDamElevation(ev) {
  newCentroid = Cesium.Cartesian3.fromDegrees(lon, lat,
    Number(svihovSlider.value));
  svihovDam.position = newCentroid;
```

```

    document.getElementById("elevationDIV").innerHTML =
svihovSlider.value - elevationDiff;
    document.getElementById("svihovCheckbox").checked = true;
    previousZ = svihovSlider.value;
}
// for mousemove slider
svihovSlider.addEventListener("mousemove", function (event) {
    if (previousZ !== svihovSlider.value) {
        changeDamElevation(event);
    }
});

```

Změna obsluhy události po kliknutí na entitu (např. bod zájmu):

```

viewer.selectedEntityChanged.addEventListener(function (entity) {
    viewer.selectedEntity = undefined; // stop opening default infoBox
    popupText.scroll({
        top: 0
    });
    popupIntroduction.classList.add("showPopup");
    try {
        const name = entity.name._value;
        const description = entity.description._value;
        popupText.innerHTML = entity.name + entity.description;
    }
    catch (e) {
        console.log("Loading entity");
    }
});

```

Obsluha slideru pro změnu průhlednosti přehrad:

```

let previousColor = damColor;
let newColor;

function changeDamOpacity(ev) {
    newColor = previousColor.withAlpha(Number(opacitySlider.value)); //
create new color from old one with different alpha
    svihovDam.plane.material = newColor;
    previousColor = newColor;
}
// for mousemove
opacitySlider.addEventListener("mousemove", function (event) {
    if (previousColor.alpha !== opacitySlider.value) {
        changeDamOpacity(event);
    }
});

```

```
    }  
  });
```

Obsluha slideru pro změnu rozlišení scény:

```
resolutionSlider.addEventListener("mousemove", function (event) {  
  if (viewer.resolutionScale !== resolutionSlider.value) {  
    viewer.resolutionScale = Number(resolutionSlider.value);  
  }  
});
```

Obsluha check boxu pro zapnutí stínů:

```
shadowsCheckbox.addEventListener("click", function (event) {  
  if (shadowsCheckbox.checked) {  
    viewer.shadows = true;  
  } else {  
    viewer.shadows = false;  
  }  
});
```

Přidání afiliace UHK do aplikace:

```
const creditUHK = new Cesium.Credit("© Univerzita Hradec Králové",  
true); // true - show credit in lightbox  
viewer.creditDisplay.addStaticCredit(creditUHK);
```

## Zadání bakalářské práce

**Autor:** Bc. Rostislav Berka

Studium: I2000215

Studijní program: B1802 Aplikovaná informatika

Studijní obor: Aplikovaná informatika

**Název bakalářské práce:** **Webová interaktivní vizualizace georeferencovaného 3D modelu**

Název bakalářské práce Web interactive visualization of georeferenced 3D model  
AJ:

### Cíl, metody, literatura, předpoklady:

Cílem práce je prozkoumat problematiku zobrazení detailních georeferencovaných 3D modelů území zahrnujících digitální model terénu i budovy ve webovém prostředí. S využitím vybrané knihovny navrhnout a implementovat interaktivní webovou aplikaci umožňující uživateli prohlížet georeferencovaný 3D model dané oblasti.

### Postup prací:

1. Prozkoumat charakteristiky a specifika vizualizace georeferencovaných 3D modelů území
2. Popsat knihovny užívané v dané oblasti
3. Navrhnout a implementovat 3D webovou aplikaci s pomocí zvolené knihovny
4. Zhodnotit a otestovat dosažené výsledky
5. Diskuze a závěr

ABDUL-RAHMAN, Alias a Morakot PILOUK, 2008. Spatial Data Modelling for 3D GIS. Berlin: Springer. ISBN 978-3-540-74166-4.

BOLSTAD, Paul, 2019. GIS Fundamentals: A First Text on Geographic Information Systems, 6th edition. Minneapolis: Eider Press. ISBN 978-1-59399-552-2.

HAYERBEKE, Marijn, 2018. Eloquent JavaScript: A Modern Introduction to Programming. New York City: Random House. ISBN 1-59327-950-7.

Zadávací pracoviště: Katedra informatiky a kvantitativních metod,  
Fakulta informatiky a managementu

Vedoucí práce: Ing. Bruno Ježek, Ph.D.

Datum zadání závěrečné práce: 26.1.2021