



# **Fyzicky realistický simulátor pražských tramvají**

Ročníková práce

Ročníková práce

Fyzicky realistický simulátor pražských tramvají včetně okolí bodovacího systému

**Gymnázium, Praha 6, Arabská 14**

Arabská 14, Praha 6, 160 00

## **ROČNÍKOVÁ PRÁCE**

**Předmět:** Fyzika

**Téma:** Fyzicky realistický simulátor pražských tramvají včetně okolí bodovacího systému

**Autor:** Martin Bykov

**Třída:** 2.E

**Školní rok:** 2024/2025

**Vedoucí práce:** Mgr. Jan Tuzar

**Třídní učitel:** Mgr. Jan Tuzar

## Prohlášení

Prohlašuji, že jsem jediným autorem tohoto projektu, všechny citace jsou řádně označené všechna použitá literatura další zdroje jsou práci uvedené.

Tímto dle zákona 121/2000 Sb. (tzv. Autorský zákon) ve znění pozdějších předpisů uděluji bezúplatně škole Gymnázium, Praha 6, Arabská 14 oprávnění výkonu práva na rozmnožování díla (§ 13) práva na sdělování díla veřejnosti (§ 18) na dobu časově neomezenou bez omezení územního rozsahu.

V Praze dne \_\_. března 2025

**X**

---

Martin Bykov, 2.E

## Poděkování

Zde bych rád poděkoval panu Eduardovi Kastnerovi, za poskytnutí informací vysvětlení specifik řízení ovládání tramvaje panu Martinovi Leubnerovi, za pomoc při nahrání zvuků během jízdy noční tramvaje. Oběma předem zmíněným lidem bych také rád poděkoval za pomoc při testování simulátoru. Dále bych rád poděkoval panu Martinovi Zahrádkovi za pomoc při hledání referenčního materiálu pro modelování tramvají.

## Obsah

Seznam použitých zkratek odborných termínů.....	7
Úvod.....	11
1. Vozidla.....	12
1.1. Modelování vozidel.....	12
1.1.1. T3R.P.....	12
1.2. Texturování nátěry vozidel.....	12
1.2.1. T3R.P.....	12
1.3. Konfigurace vozidel.....	15
2. Implementace vlastní grafické infrastruktury.....	19
2.1. Okno grafické rozhraní.....	19
2.1.1. Postup implementace.....	19
2.1.2. Kompatibilita mezi operačními systémy.....	19
2.1.3. Infrastruktura vykreslování objektů na obrazovku.....	20
2.1.4. Grafické rozhraní.....	21
2.2. Vykreslování modelů.....	22
2.2.1. Načítání vykreslování modelů, systém GMS.....	22
2.2.2. Načítání vykreslování textur modelů.....	23
2.2.3. Vykreslování průhledných částí modelů.....	23
2.3. Vykreslování okolí.....	24
2.3.1. Vykreslování kolejí.....	24
2.3.2. Vykreslování silnic, země, ulic, kolejových svršků stěn.....	24
2.3.3. Vykreslování zastávek a návěstidel.....	25
2.3.4. Vykreslování budov památek.....	25
3. Prvky předpisy ovládání simulátoru.....	26
3.1. Pohyb po trati návěsti.....	26
3.2. Výhybky.....	27
3.2.1. Elektrické.....	27
3.2.2. Rádiové.....	27
3.2.3. Zajišťovací výhybky.....	27
3.2.4. Přehazování výhybek.....	27
3.3. Návěstidla.....	28
3.3.1. Výhybková návěstidla.....	28
3.3.2. Křižovatková návěstidla.....	29
3.3.3. Předvěsti křižovatek křižovatek návěstidel.....	29
3.4. Jízdní řád.....	30
3.4.1. Normální zastávky.....	30
3.4.2. Zastávky na znamení.....	30
3.4.3. Kontrolní body.....	30
3.4.4. Konečné zastávky.....	30
3.5. Identifikace tramvaje.....	31
4. Dodatečné nástroje formáty jejich souborů.....	32
4.1. Nástroj na úpravu mapy scénářů.....	32
4.1.1. Základní informace.....	32

4.1.2. Horní lišta.....	33
4.1.3. Menu úprav.....	34
4.1.4. Plátno úprav.....	34
4.1.5. Tvorba linky.....	35
4.2. Formát souboru okolí.....	37
4.3. Formát souboru scénáře.....	41
4.4. Nástroj na vytváření hlásiče.....	43
4.5. Formát souboru hlásiče.....	44
5. Implementace simulací.....	46
5.1. Infrastruktura působení sil na vozidla.....	46
5.2. Tíha.....	46
5.3. Tření, valivý odpor prokluz kol.....	47
5.4. Síla působení motorů rychlost vozidla.....	48
5.5. Síly oblouku vykolejení.....	49
5.6. Aerodynamická odporová síla.....	50
5.7. Elektrické systémy vozidel celková práce vozidla.....	50
5.8. Osvětlení.....	51
5.9. Stíny.....	54
5.10. Simulace zvuků.....	56
5.10.1. Simulace zvuku motoru brzd.....	56
5.10.2. Simulace zvuku jízdy.....	57
5.10.3. Simulace jiných zvuků.....	57
5.10.4. Simulace hlášení.....	57
5.11. Výpočet srovnávání emisí.....	58
5.11.1. Výpočet emisí automobilové dopravy.....	58
5.11.2. Výpočet emisí tramvaje.....	58
5.11.3. Výpočet ušetřených emisí.....	58
6. Simulace počasí jeho účinků.....	59
6.1. Infrastruktura systému počasí.....	59
6.2. Denní noční cyklus.....	59
6.3. Počasí.....	59
6.3.1. Vítr.....	59
6.3.2. Déšť.....	59
6.3.3. Bouřky.....	59
6.3.4. Mlha.....	59
6.4. Počasí závisující na ročním období.....	60
6.4.1. Sníh.....	60
6.5. Roční období.....	60
6.5.1. Jaro léto.....	60
6.5.2. Podzim.....	60
6.5.3. Zima.....	60
7. Modelování okolí.....	61
7.1. Zastávky.....	61
7.2. Standardní budovy.....	61
7.2.1. Typy budov.....	61

7.2.2. Systém obarvení budov.....	64
8. Systém bodování.....	65
8.1. Seznam akcí za které se body odečítají.....	65
8.2. Seznam akcí za které se body přičítají.....	65
8.4. Výpočet výsledné známky.....	65
9. Další rozvoj projektu.....	66
9.1. Modelování dalších vozidel.....	66
9.2. Modelování dalších tramvajových tratí.....	66
9.3. Vylepšení modelu osvětlení.....	67
9.4. Přidání řek mostů.....	67
9.5. Přidání opravdového terénu.....	67
9.6. Přidání nastavení trakčního vedení stahování pantografu.....	67
Závěr.....	68
Použité technologie.....	69
Seznam obrázků.....	70
Seznam tabulek.....	70
Citace.....	71

## Seznam použitých zkratk odborných termínů

**ASCII** Kódování znaků zaujímající vždy právě byte. Podporuje pouze anglickou abecedu, číslice, pomocné znaky typu mezer závorek či kontrolní znaky.

**Abstrakce** Zabalení obvykle několika systémových („nízko-úrovňových“) funkcí do jiné funkce či objektu za účelem lehčího vývoje použití daných funkcí.

**Anti-aliasing** Technologie zahlazování pixelu pro zhlazení viditelného přechodu mezi pixely

**APEX** Společnost vyvíjející palubní počítače, hlásiče rádiové stanice. Jejich název se používá jako přezdívka palubního počítače tramvaje řad T3R.P, T3R.PV, T3R.PLF, T6A5 KT8D5.RN2P.

**API** Application programming interface – soubor funkcí tříd pro přístup programu nějaké infrastruktuře funkcím

**Atenuace** Zeslabení vlnění při vzdálenosti (při průchodu jakéhokoliv materiálu). grafickém programování znamená hodnotu zeslabení světla určitou vzdáleností.

**Bitové pole** Místo uložení, zpravidla několika, Booleanových hodnot. Bitové pole ukládá hodnoty které mají vliv na pracování programu, jinými slovy flagy.

**Boolean** Hodnota která může nabývat pouze hodnoty či a je jí možné uložit 1 bitu paměti.

**Buffer** Názvosloví OpenGL pro úložný prostor aneb vyhrazená paměť pro ukládání dat na grafické kartě. Existuje vícero typů bufferů (vertexový, indexový, shaderový, framebufferový apod.).

**DPP** Dopravní podnik hlavního města Prahy (také známý jako Dopravní podnik hl. m. Prahy, a. s.), jediný provozovatel tramvají typu T3R.P Praze.

**Dvojkový doplněk** Standardní způsob ukládání záporných hodnot binární reprezentaci celých čísel. První bit určuje hodnotu jako  $-2^b$  kde je počet bitů čísla.

**Endianita, little-endian, big-endian** Pořadí bytů při ukládání čísla. Ve formátu little-endian je na konci byte nejmenší hodnotu. Ve formátu big-endian je na konci byte největší hodnotou. Endianita (*angl. Endianness*) určuje jaký těch dvou formátů je použit.

**ev. č.** Evidenční číslo vozidla, užívané pro identifikaci vozidla dopravcem

**Flag** Booleanova hodnota mějící vliv na průběh programu.

**Fragment** Pixel, bod ve světě – pro každý takový bod je spuštěn tzv. fragmentový shader, který vypočítává jeho barvu.

**Framebuffer** Speciální buffer napojenou texturou použitý pro vykreslování mimo obrazovku.

**GLAD** Knihovna pro načítání funkcí všech variant API OpenGL.

**GLFW** Knihovna pro vytváření okna detekce vstupu klávesnice či myši. Plným názvem OpenGL Framework.

**GLM** Knihovna pro práci strukturami jako například vektory či matice, často OpenGL užíváné.

**GLSL** Jazyk, podobný jazyku avšak bez velkého počtu funkcí, použitý shaderech OpenGL.

**Hexadecimální kód** Způsob kódování RGB barev ve rozsahu od až 255 (tzv. 8-bitové kombinování,  $2^8 - 1 = 255$ ). Hodnoty jsou vyjádřené šestnáctkové soustavě (od 00 do ff).

**Index** Identifikační číslo vertexu. Jsou používány pro eliminaci duplicitních vertexů. Zamísto duplikování bude tak použit jeden vertex na který se bude prostřednictvím indexu ukazovat.

**Instancování** Anglicky „instanced rendering“, též instancing neboli „instancování“. Jedná se způsob vykreslování stejného meshe několikrát jediným příkazem nastavením potřebné proměnlivé informace (posunem, barvou apod.) za pomoci UBO nebo SSBO bufferů před vykreslováním. shaderu je přístupná informace který objekt se jedná.

**ISO** Mezinárodní organizace pro normalizaci („International Organization for Standardization“) je organizací která stanovuje jasně daná pravidla pro, mimo jiné, barvy formy varovných symbolů.

**IZS** Integrovaná záchranná služba. tomto textu je zkratka použita jako označení pro jakékoliv pohotovostní vozidlo, to jak policejní, hasičské či nemočnické.

**Leaflet** Knihovna vytvořená jazyce JavaScript používaná pro jednoduché vykreslování mapových podkladů.

**Materiál** Soubor vlastností pro daný mesh. Mezi vlastnosti patří například textura, barva, lesk, odrazivost, průhlednost atd.

**Mesh** Odborný termín pro veškerá data (vertexy, obličeje) modelu nebo jeho části. Nejčastěji souvislý.

**MHD** Zkratka pro městskou hromadnou dopravu.

**MSAA** „Multisample anti-aliasing“ – anti-aliasing technologie poskytnutá technologií OpenGL

**Normalizace** Normalizace vektoru je převedení vektoru takovým způsobem, aby udával stejný uhel jeho délka činila hodnotu 1.

**Normál** Vektor kolmý na daný obličej.

**Nulový znak** Jinak známý jako nulový terminátor, tento znak hodnotě (0 neoznačuje číslici, nýbrž hodnotu bytu) často slouží jako zakončení textových řetězců.

**OpenGL** Zkrácená forma Open Graphics Library, API použité pro infrastrukturu simulátoru zabývající se vykreslováním objektů na obrazovku.

**OpenStreetMap** Zdarma open-source mapa, použita nástrojích simulátoru.



**PID** Pražská integrovaná doprava, systém vytvořený organizací ROPID (Regionální organizátor PID) který je zodpovědný, mimo jiné, za nový jednotný nátěr vozidel.

**Průhlednost** tomto textu se průhlednost udává jako viditelnost procentech, kde procent znamená že je daný objekt neviditelný zatímco 100 procent znamená že je zcela neprůhledný.

**Pixel** Nejmenší jednotka počítačové grafice – označuje jedno světlo na monitoru.

**RGB** Název formátu kódování barev – znamená „Red, Green, Blue“.

**Shader** Program napsaný jazyce GLSL spuštěný na grafické kartě během procesu vykreslování. Je používán pro výpočet určení barev pixelů, změn pozic bodů či jiných výpočtů které je potřeba udělat pro každý pixel zvlášť.

**STBI** Knihovna pro načítání obrázků kolekce jednosouborových knihoven STB. Plným názvem „STB Image“.

**SSBO, UBO, Uniformy** Prostředky pro komunikaci shaderu programu. Uniformy mají nejmenší kapacitu, uniformová úložiště (UBO, „Uniform buffer object“) větší SSBO („Shader storage buffer object“) největší<sup>1</sup>.

**Teselace** „parkétování“ – pokrytí povrchu různými geometrickými tvary

**Textura** Obrázek či jeho část určující (nebo ovlivňující) výslednou barvu jakéhokoliv fragmentu.

**Texturové parcely** Název pro samostatný obdélník daným materiálem pro který jsou programu vygenerovány vertexové indexové buffery. Texturové parcely nemusí povinně mít textury, stejně jako materiály mohou mít pouze barvu.

**Unixový čas** Počet sekund uplynulých od 1. ledna 1970, 00:00:00 koordinovaném světovém čase.

**UTF-8** Písmenná sada obsahující celou latinskou abecedu, číslice většinu znaků diakritikou užívanou českém jazyce. Každé písmeno je uloženo 1 bytu neboli bitech.

**Vertex** Bod jakéhokoliv modelu. Může mít víc než jen polohovou informaci – může také obsahovat barvu, číslo materiálu, normály obličeje kterého je součástí atd.

**Vykreslovací příkaz** Příkaz který program posílá grafické kartě. Hlavní cíl jakékoliv optimalizace je minimalizace jeho počtu.

**WAV** Zkratka pro Waveform Audio File Format. Jedná se nekomprimovaný formát zvukových souborů.

**Z-Boj** (angl. *Z-fighting*) Neurčitost mezi dvěma texturami které jsou ve stejném místě, výsledek tohoto je prolínání obou textur na zasaženém místě.

---

<sup>1</sup> UBO má dle standardy OpenGL 4.5 CITACE WIKI

## **Anotace**

Cílem tohoto projektu je implementace realistického simulátoru pražských tramvají ze pomocí jazyka C++ grafické rozhraní OpenGL. Jsou simulované interakce vozidla kolejí, rychlostní limity, výhybky, osvětlení, elektrické systémy tramvaje, efekt počasí na ročních obdobích na předem zmíněné prvky i realistický dynamický zvukový systém při rozjezdu zatáčení. Současně je simulován pouze typ T3R.P na lince 13 úseku Olšanské hřbitovy – Museum – Čechovo náměstí odbočce na obratiště Zvonářka.

## **Abstract**

The goal of this project is the implementation of physically realistic simulator of trams in Prague using the C++ programming language and the OpenGL graphics API. Among the simulated parts are the interactions of the vehicle and the track, the speed limits, switches, lighting, the electrical systems of the tram, the effect of the weather on the mentioned parts as well as realistic dynamic sound system during the acceleration and turning of the tram. Currently, only the T3R.P tram type is simulated and only on line 13, which goes from Olšanské hřbitovy to Čechovo náměstí via Museum and the branch line to the Zvonarka terminus.

## **Аннотация**

Цель этого проекта это создание физически реалистического симулятора пражских трамваев помощью языка C++ графической «библиотеки» ОпенГЛ. Симулированные будут влияния трамвая рельс, освещение, электрические системы трамвая, влияние погоды на все симуляции симуляция звуков трамвая. проекте симулирован только тип трамвая ТЗР.П на линии 13 между остановкой Olšanské Hřbitovy остановкой Čechovo náměstí ответвление остановке Zvonářka.

## Úvod

Cílem tohoto projektu je implementace fyzicky realistického simulátoru pražských tramvají (současně simulace obsahuje pouze tramvaj T3R.P) za pomoci vlastní grafické infrastruktury. Simulátor obsahuje simulaci vzájemného působení (tření, zvuk) mezi kolejemi koly tramvaje, simulaci výhybek, simulaci deformace silničních vozidel při nárazu, simulaci elektrických systémů tramvaje i vliv počasí na předem zmíněné prvky simulace. Projekt bude možné použít nejen jako simulátor pro trénování dalších řidičů tramvaje, ale vizualizace simulátoru bude možné také použít jako fyzikální demonstraci simulovaných prvků.

## 1. Vozidla

### 1.1. Modelování vozidel

#### 1.1.1. T3R.P

Modelování vozidla T3R.P bylo poměrně nenáročné. Modelování karoserie formy tramvaje bylo jednoduché důvodů kvalitního referenčního materiálu, dosaženého fotogalerie vozidel MHD<sup>[3][4][5]</sup>. Nejtěžší částí pro modelování byly podvozky tramvaje pantograf (sběrač elektrického proudu trakčního vedení) důvodů nedostatku kvalitního zdrojového materiálu pro právě tyto části vozu.<sup>[6]</sup> Pro pantograf byl později nalezen technický výkres, bohužel avšak pro původní, nerekonstruovanou řadu T3<sup>[7]</sup>, pro podvozek byl nalezen 3D model podobného typu T3.SUCS.<sup>[8]</sup>

Řada T3R.P byla vybrána důvodu její rozšířenosti ve flotile DPP, kde je 337 vozů řady T3R.P 782 tramvají celkem, což celkem tvoří 43% procenta všech tramvají.<sup>[9]</sup>

### 1.2. Texturování nátěry vozidel

Texturování vozidla probíhalo ve fázích. Nejdříve byli vytvořeny jednotlivé materiály textury pro nich následně byli každé části modelu materiály přiřazeny. Piktogramy varovné značky byli vytvořeny přidány zvlášť. Pro zachování realismu byli mezinárodně standardizované varovné znaky ISO importovány externích zdrojů.<sup>[10][11]</sup>

#### 1.2.1. T3R.P

Tramvaj simulátorů má několik různých nátěrů. Jsou přidány oba dva běžné nátěry vozů DPP, to červeno-krémový vinovo-stříbrný (užívaný je sice primárně vozy typu T3R.PLF, je ale také použit vlečnými vozy typu T3R.P je možné zahlédnout tyto vozy samostatně během testovacích či zvláštních jízd). Všechny nátěry sdílí společné zabarvení interieru.<sup>[3]</sup>

Hodnot barev schémat byli převzaty veřejně dostupných fotografií tramvají typu T3R.P.<sup>[3]</sup> Barvy pro moderní nátěr PID byly uvedeny grafickém manuálu jednotného vizuálního stylu PID roku 2024.<sup>[12]</sup> následující tabulce (*Tabulka 1*) jsou uvedeny názvy hexadecimální RGB kódy všech barev přítomných na modelu tramvaje.

Červená (nátěr)	#e72e19
Krémová (nátěr)	#f2ebdf
Vínová (nátěr)	#701021
Stříbrná (nátěr)	#7f7d7e
Červená, jednotný PID (nátěr)	#bb1e10 / RAL 3020
Šedá, jednotný PID (nátěr)	#b0b0a9 / RAL 7038
Hnědá (nátěr)	#6b5646
Bílá (interiér, laminát – světlejší části)	#eeeeee
Šedá (interiér, laminát – tmavší části), Železná, Skleněná	#aaaaaa
Tmavě šedá (interiér – šedé železné části)	#303030
Tmavě železná	#454545
Červený plast na sedačkách	#ff0000
Šedý plast na sedačkách	#454545
ISO 3864 – Žlutá (varování)	#f9a900
ISO 3864 – Červená (zákaz, požární technika)	#9b2423
ISO 3864 – Bílá (podklad)	#ecece7
Žlutá (tlačítka, tyče pro držení)	#ffce00
Oranžová (stroje pro označení jízdních dokladů)	#ff9000
Zelená (dveřní tlačítka)	#00c16e
Tmavě červená (lampy, zadní reflektory)	#800000
Tmavě oranžová (světla pro odbočování)	#a15a00
Černá	#000000
Modrá	#000080

*Tabulka 1: Seznam barev použitých modelu.*

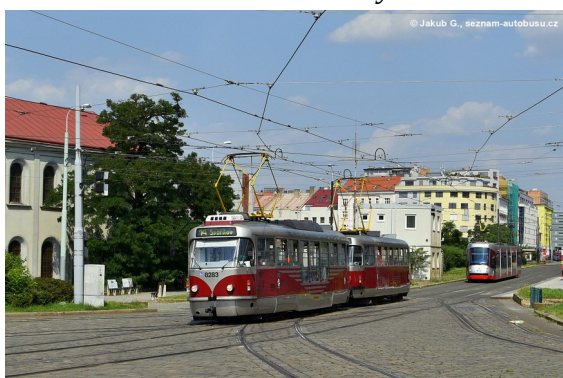
Textury pro nerovnosti interiéru vozidla byli spojením šedého pozadí vrstvy 90 procentní průhledností na kterou byla následně nanесena šedá barva barevného elektronického šumu který byl následně převeden na černobílou škálu. Nerovnosti textury interiéru byli nanесeny speciálním štětcem editačním programu Krita pod názvem „Texture Crackles“. Textury pro světelná zařízení byly vytvořeny nastavením 70 procentní průhlednosti na požadovanou barvu. Pro čiré sklo byla použita železná barva, to 90 procentní průhlednosti.



Obrázek 1: Fotografie vozů T3R.P (ev. č. 8234 8235) DPP nedaleko Sídliště Řepy<sup>[3]</sup>



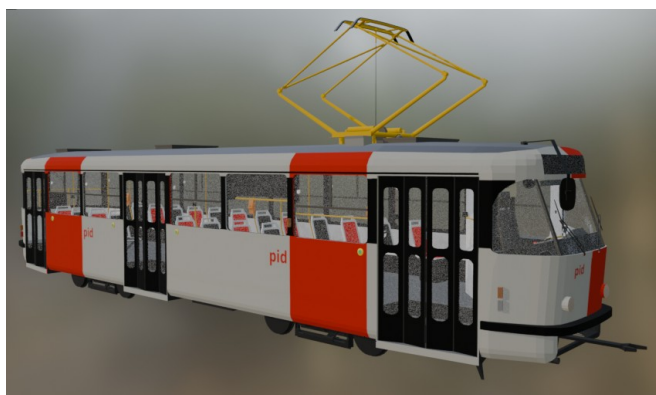
Obrázek 2: Červeno-krémový nátěr na modelu tramvaje typu T3R.P



Obrázek 3: Vůz T3R.PLF (ev. č. 8283) vlečný vůz T3R.P (ev. č. 8160) na Palmovce<sup>[3]</sup>



Obrázek 4: Vínovo-stříbrný nátěr nanесený na modelu typu T3R.P



Obrázek 5: Umělecké zobrazení nátěru PID vytvořeného dle grafického manuálu PID

### 1.3. Konfigurace vozidel

Vozidla je možné přidávat konfigurovat za pomoci konfiguračního souboru umístěného ve složce programu. konfiguračním souboru jsou uvedeny nejen fyzikální parametry vozidla – rozměry skříně vozidla ze kterých je pak vypočtena plocha stěn skříně vozidla, která je pak použita při výpočtech síly větru, odkaz na zvuky jízdy vozidla, hmotnost vozidla kapacita vozu, ale také grafické parametry – barva panelů, seznam nátěrů, propojení meshů modelů podvozků které je posouvají, posunu kamery řidiče či prvky ovládací – posun rozměr tzv. „tlačítkových zón“, neboli zón při kliknutí na které se daná akce spustí. Každá hodnota je uložena následujícím formátu:

NÁZEV\_HODNOTY=HODNOTA

*NÁZEV\_HODNOTY* i *HODNOTA* jsou řetězce ve formátu UTF-8. Řetězce smí obsahovat všechny znaky UTF-8 kromě znaku „rovná se“, čárky středníku. Mezery jsou započítány jako součást hodnoty. Na každém řádku je přítomna právě jedna hodnota. Má-li konfigurační soubor pro nějaký parametr více hodnot, jsou jednotlivé hodnoty oddělené čárkou. Obsahuje-li parametr více vnitřních parametrů, jsou odděleny středníkem. Příklady formátování několikanásobných hodnot souboru jsou uvedeny zde:

NÁZEV\_HODNOTY=HODNOTA, HODNOTA

NÁZEV\_HODNOTY=HODNOTA, HODNOTA; HODNOTA2, HODNOTA2

V následujících tabulkách jsou uvedeny všechny povinné hodnoty pro vůz T3R.P<sup>[13]</sup> všechny povolené konfigurační hodnoty. tabulkách nejsou uvedeny hodnoty souborových cest, jelikož jsou relativní mohou se jednoduše měnit. Konfigurační soubor musí povinně všechny uvedené hodnoty obsahovat. Souborové cesty jsou uloženy souborům ukládajícím zvuky rozjezdu brzdění, jízdy, uzávěru otevření dveří, zvonění znamení řidiči. Ke každému zvuku je také uložen zvuk opačný. Zvuk znamení řidiči zvuk signalizace uzávěru dveří je typu T3R.P totožný.<sup>[2]</sup>

V konfiguračním souboru je potřeba použít tečku, nikoliv čárku, jako oddělovač desetinných míst. všech hodnot kromě nátěrů se bere pouze poslední definovaná hodnota. nátěrů se základní nátěr neuvádí. Prázdné řádky jsou povoleny. Komentáře jsou řádky které začínají na znak „#“ jsou přeskočeny. Nepoužité hodnoty jsou nastavené na prázdné řetězce.

Udává-li hodnota informaci spjatou polohou má-li číselné hodnoty za sebou, jsou interpretovány jako hodnoty pro osy X, Y, v tomto pořadí. Model by měl být nastaven tak, že osa X+ směřuje do levé strany vozidla, osa Y+ na vrch vozidla osa Z+ čelu vozidla. Má-li hodnota číselné hodnoty za sebou, udává poslední číslice vodorovné otočení po směru hodinových ručiček ve stupních. Odsazení je vždy udáno od středu článku ve kterém se položka nachází.

Uživatelský název parametru	Název hodnoty	Hodnota
Hmotnost vozidla (v kilogramech)	MASS	17 300
Délka vozidla (v metrech)	LENGTH	15.104
Šířka vozidla (v metrech)	WIDTH	2.5
Výška skříně vozidla (v metrech)	HEIGHT	3.06
Počet sedacích míst ve vozidle	SEATS	22
Maximální počet stojících cestujících	STANDING	69
Počet motorů	MOTORS	4
Výkon motoru (v kilowattech)	POWER	40
Maximální provozní rychlost (v km/h)	SPEED	65
Název podvozků modelu	BOGIENAMES	T3 bogie 1,T3 bogie 2
Přípony náprav podvozků (musí být 2)	BOGIEWHEELS	wheels F,wheels B
Seznam meshů přidělených každému podvozku.	BOGIEMESHES	Prázdná hodnota.
Odsazení podvozků, pouze na vodorovných osách X, Z	BOGIEOFFSETS	0.0,3.225;0.0,-3.225
Odsazení podvozků na trati	BOGIETROFFSETS	2.5,8.95
Materiál modelu na který se budou nátěry aplikovat	LIVERYMAT	Material.paint
Nátěry	LIVERY	PLF,soubor textury PID,soubor textury
Odsazení pohledu řidiče	CAMERAOFFSET	0.2,2.3,5.9
Otáčky motoru	MOTORRPM	1750
Hodnota převodovek	GEARRATIO	7.43
Průměr kol metrech	WHEELDIAMETER	0.6
Koeficient aerodynamického odporu	AEROCOEF	1.8

Tabulka 2: Konfigurační hodnoty pro typ T3R.P<sup>[13][14]</sup> CITACE ENGINEERINGTOOLBOX



Název ovládacího prvku	Název hodnoty	Dodatečná hodnota (TYPE)	Hodnota posunu (OFFSET)	Hodnota rozměru (SIZE)
Přepínač řízení	STARTER		-0.2,1.5,5.9	
Pískovač	SANDER		0.4,1.0,5.9	0.1,0.3,0.3,0.0
Odhlášení zastávky	STOP		TODO	
Zvonek	BELL		TODO	
Dveře řidiče (zde první dveře vozu)	DDOOR		TODO	
Tísňové tlačítko	EMERGENCY		TODO	
Otevření dveří	OPENDOOR	L	TODO	
Uvolnění dveří	FREEDOOR	R	TODO	
Zavření dveří	CLOSEDOOR	M	TODO	
Typ ovládání dveří	DOORS	switch *		
Nouzová brzda	BRAKEE		TODO	0.1,0.2,0.1,0.0
Ruční odbrzdění brzdy (zde čelist'ové/mechanické)	BRAKE1		TODO	
Osvětlení denní	LIGHTDAY	L	TODO	
Osvětlení noční	LIGHTNIGHT	R	TODO	
Typ ovládání osvětlení den/noc.	LIGHT	switch *		
Osvětlení, přepnutí režimu	LIGHTMODE		TODO	
Vypnutí osvětlení kabině	CLIGHTON	R	TODO	
Zapnutí osvětlení kabině	CLIGHTOFF	M	TODO	
Typ	CLIGHTTYPE	switch *		

Tabulka 3: Posunové hodnoty pro ovládání vozidla typu T3R.P

\* – Buď hodnota *switch*, může být také *buttons*. Switch označuje přepínač – přepínač má polohy označené *L R P*. a jsou na levé straně, je střední poloha *R*, jsou na pravé straně. Krajní písmena *K*, jsou od středu dále než vnitřní písmena a *R*. Písmen může být nastaveno několik. Pokud je hodnota *buttons*, jsou všechny ovládací prvky pro dané akce zvlášť.

Základní hodnota pro rozměr ovládacích prvků je *0.1,0.1,0.1,0.0*. (krychle straně 1dm obsahu právě 1dm<sup>3</sup> bez žádné rotace).

Uživatelský název parametru	Název hodnoty
Soubor modelu, příponou	MODEL
Zvuk otevírání dveří	SOUNDOPENDOOR
Zvuk zavření dveří	SOUNDCLOSEDOOR
Zvuk otevírání dveří do kabiny	SOUNDOPENCABIN
Zvuk zavírání dveří do kabiny	SOUNDCLOSECABIN
Zvuk rozjezdu vozidla	SOUNDSTART
První zvuk jízdy vozidla	SOUNDRIDE1
Převrácený první zvuk jízdy	SOUNDRRIDE1
Druhý zvuk jízdy vozidla	SOUNDRIDE2
Převrácený druhý zvuk jízdy	SOUNDRRIDE2
Třetí zvuk jízdy vozidla	SOUNDRIDE3
Převrácený třetí zvuk jízdy	SOUNDRRIDE3
Zvuk brzdění	SOUNDBRAKE
Zvuk přejezdu výhybky	SOUND SWITCH
Zvuk dveřní signalizace znamení řidiči	SOUNDINFO

Tabulka 4: Další povinné konfigurační hodnoty

## 2. Implementace vlastní grafické infrastruktury

### 2.1. Okno grafické rozhraní

#### 2.1.1. Postup implementace

Implementace grafické infrastruktury probíhalo několika stádiích. Nejdříve byly implementovány základní abstrakce systémových funkcí, následně bylo implementované osvětlení, stíny, načítání 3D modelů, okolí včetně tratí, fyzické simulace jako poslední také návěstidla, odjezdové tabule či panely na vozidlech.

#### 2.1.2. Kompatibilita mezi operačními systémy

Pro kompatibilitu mezi různými platformami byl vybrán standardní jazyk C++, konkrétněji verze C++20. Tento jazyk je nejen možné na všech operačních systémech používat, sám je často daných operačních systémech použit.<sup>[15]</sup>

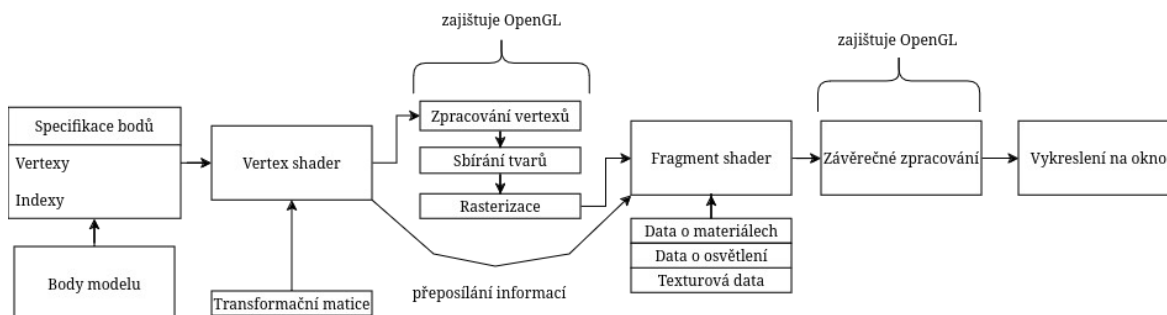
Z téhož důvodu bylo vybráno API OpenGL, které je podporované na všech nejrozšířenějších operačních systémech – Windows, MacOS Linux. Na něj navazují všechny ostatní technologie při vývoji použité, jako třeba GLFW („OpenGL Framework“), GLEW („OpenGL Extension Wrangler“) GLM („OpenGL Mathematics“).

Kompatibilita systému vypisování barevného textu byla docílena rozdělením funkce přebarvení na části – část pouze pro GNU/Linux část pouze pro Windows. Pro část GNU/Linux byly využity tzv. ANSI únikové kódy<sup>[16]</sup>.

Současně kompilační infrastruktura programu, založena na systému Cmake, podporuje pouze operační systémy Windows (přes překladač MinGW) Linux (přes překladače GCC nebo Clang). Podpora platformy MacOS by po menších změnách měla být možná.

Editory pro hlásič, scénář linku jsou napsány jako webové stránky JavaScriptu. Toto znamená že je možné editory spustit na jakémkoliv zařízení monitorem. Na mobilních zařízeních je sice spuštění možné, není to avšak oficiálně podporované povede degradovanému uživatelskému prostředí.

### 2.1.3. Infrastruktura vykreslování objektů na obrazovku



Obrázek 6: Proces vykreslování modelu grafickým rozhraním OpenGL 4.5

OpenGL používá jistý proces (viz obrázek 6) při vykreslování modelů.<sup>[17]</sup>

Jako první čte vstupní data nahrané programem do paměti grafické karty počítače – vertexy (body) indexy (pořadí bodů). Dále jsou nastaveny shadery – vertexový shader fragmentový (či pixelový) shader. Vertexový shader je spuštěn pro každý bod na modelu. Fragmentový je spuštěn jednou pro každý pixel na modelu. vertexového do fragmentového shaderu je možné posílat informaci uvedenou bodech. Program může také se shadery komunikovat za pomoci SSBO UBO. Přes SSBO je možné posílat pouze skalární hodnoty, 2-rozměrné vektory 4-rozměrné vektory. 3-rozměrné vektory důvodu paměťové struktury OpenGL posílat nelze<sup>2</sup>. téhož důvodu je potřeba vždy nastavit 16-bitové zarovnání paměti.<sup>[18\*]</sup>

Po načtení dat je spuštěn nejdříve shader vertexový, následně pak fragmentový. Ve vertexovém je určena pozice bodu přeposílána informace do shaderu fragmentového.<sup>[17\*][19]</sup>

Ve fragmentovém shaderu jsou implementovány veškeré výpočty týkající se osvětlení barev. Ve vertexovém shaderu jsou implementovány veškeré posuny změny vertexů meshů. Geometrové shadery (angl. Geometry shaders), které mohou přidávat nové vertexy, nejsou simulátoru použity.<sup>[17\*]</sup>

V rámci optimalizací vylepšení kvality vykreslování byly použity funkce API OpenGL „face culling“ neboli nevykreslování zadních stran stěn s tím související funkce „depth testing“ (hloubkové testování), která vykresluje pouze stěny které je pohledu uživatele vidět.<sup>[20 p.174-181]</sup> Pro vylepšení kvality byla použita funkce „anti-aliasing“ (konkrétněji technologie MSAA, jenž funguje na základě vyprůměrování sousedních pixelů) která zhlazuje strmý přechod mezi jednotlivými pixely.<sup>[20 p. 264-271]</sup>

2 Dle standardy OpenGL je možné posílat 3-rozměrné vektory, žádné implementace to jednak nepodporují často to vede neřešitelným problémům.<sup>[18]</sup>

#### **2.1.4. Grafické rozhraní**

Grafické rozhraní bylo implementováno za pomoci knihovny Dear ImGui. Použití grafického rozhraní (GUI) simulátoru je možné rozdělit na části: hlavní menu, simulační na pauzovací.

Při spuštění programu se objeví hlavní menu. hlavního menu je možné začít jízdu simulátoru, změnit nastavení simulátoru či ho také opustit. Menu nastavení obsahuje jednotlivé možnosti pro změnu parametrů simulace či změnu mapy, linky či hlásiče. Změna vozidla či nátěrů není zatím nastaveních podporována.

Během režimu simulátoru je dostupné simulační rozhraní. Rozhraní nese většinou informativní charakter – informuje stavu scénáře, počtu nasbíraných bodů během jízdy, současnou rychlost tramvaje o jiných potřebných během jízdy informacích. Je také přítomna funkce vypínání zapínání grafického rozhraní za pomoci klávesy.

V rozhraní při menu pauzy je možné simulátor opustit, vrátit se do hlavního menu či změnit nastavení simulátoru.

## 2.2. Vykreslování modelů

### 2.2.1. Načítání vykreslování modelů, systém GMS

Modely jsou obvykle rozdělené na několik částí, odborně nazývaných meshe. Toto rozdělení je provedeno autorem. Simulátor podporuje pouze formát GLTF, který je založen na formátu JSON. Nejdříve jsou načteny všechny materiály modelu, případně textury daného materiálu. Dále je načtena hierarchie bodů jejich transformace. Nakonec program načítá jednotlivé meshe modelu. Pro každý mesh jsou načteny všechny jeho vertexy, jejich normálové vektory, texturové souřadnice určující jakou část textury má daný vertex použít, indexy následně jiné parametry vertexu, například indexy kostí jejich koeficienty vlivů.

Všechny materiály jsou uloženy globálním úložišti, označovaném programem jako „Global Material Store“, zkráceně GMS. Samotné meshe ukládají pouze odkaz na danou položku GMS. Materiály ukládají svoje parametry pro model světla svojí texturu, pokud jí nemají pak svojí barvu. Za pomoci systému GMS je zajištěna možnost změny materiálu meshe – každý materiál má uložené také jeho varianty. Pro jednodušší identifikaci „variant“ materiálu má každý materiál také identifikační text – pro standardní variantu je použit speciální, předem definovaný text. Varianty materiálu jsou přidány uživatelem, program automaticky načítá pouze standardní variantu. Systém GMS také eliminuje duplikaci totožných materiálů. Totožné materiály jsou detekovány dle souborové cesty textury. Během práce programu není možné měnit materiály meshe, je možné avšak využít variantového systému GMS. Každý materiál může mít přiřazené varianty. Tento systém změny variantů je simulátoru užíván například nátěrů vozů či světél domech.

Vykreslování modelů probíhá několika stupních. Vertexy indexy jsou automaticky poslány do fragmentového shaderu. Parametry materiálů jsou uloženy systémem GMS do SSBO, který následně informaci přepoše do fragmentového shaderu, který pro každý pixel vypočítává barvy aplikuje dodatečné efekty, například světla stíny.

### 2.2.2. Načítání vykreslování textur modelů

Během načítání souboru modelu se taky načítají textury jednotlivých modelů. Každý materiál má přiřazenou právě texturu každý vertex meshe má přiřazen právě jeden materiál. Celkem ve stejný čas je možné jednomu fragmentovému shaderu využít právě 32 textur<sup>3</sup>, některé těchto texturových pozicí jsou ale rezervovány pro použití systémem stínů, zaujímající současně slot pro každé světlo které má simulovaný stíny, tedy celkem sloty, nebo systém počasí který současně zabírá slot pro detekci střech. Pozice pro textury systému vykreslování stínů se vždy počítají od konce. Materiály nemusí mít pouze texturu – mohou mít také nastavenou barvu při vykreslování mohou použít oboje. Tato hodnota je vyjádřena parametrem materiálu  $\vec{F}_g = m_t \times \vec{g}$  která určuje procentuální podíl textury na výslednou barvu materiálu. Je-li tato hodnota rovna nule, neboli nemá-li materiál žádnou přiřazenou texturu, nezabírá shaderu žádný texturový slot. tohoto plyne limitace simulátoru – každý mesh může obsahovat nanejvýše 27 texturovaných materiálů.

### 2.2.3. Vykreslování průhledných částí modelů

Vykreslování průhlednosti není zcela jednoznačnou věcí. Pro vykreslování všech průhledných částí za sebou, to nezávisle od pořadí, je potřeba využít technologii OIT, celým názvem „order independent technology“ neboli česky průhlednost nezávislá na pořadí.<sup>[21\*]</sup>

Existuje několik technologií OIT – simulátor používá technologii „weighted-blended“, vynalezenou roce 2013 společností NVIDIA. Technologie funguje na principu vykreslování průhledných neprůhledných objektů zvlášť, jejich následné kombinování vykreslení. Pro podporu anti-aliasingu jsou avšak simulátoru implementovány změny.<sup>[21\*]</sup>

Framebuffer, na které se průhledné neprůhledné části vykreslují, jsou nastaveny na tzv. „multisamplované“, ve kterých každý pixel může mít několik hodnot. Toto pomáhá plynulejším přechodům mezi hranicemi objektů. Tato změna zvyšující kvalitu obrazu avšak požaduje přidání dalšího kroku, to překopírování již kombinovaného multisamplovaného bufferu na buffer nemultisamplovaný, který je následně vykreslen na obrazovku.<sup>[20 p.268]</sup>

Neprůhledné objekty jsou vykreslovány do jejich framebufferů beze žádných změn. průhledných objektů se provádí dodatečné výpočty ukládají se hodnoty – upravená zprůměrovaná barva výsledná průhlednost daném fragmentu.

---

3 Většina moderních grafických procesorů podporuje 32 slotů či více, samotné API OpenGL avšak určuje minimální počet textur dostupných ve fragmentovém shaderu jako 16.

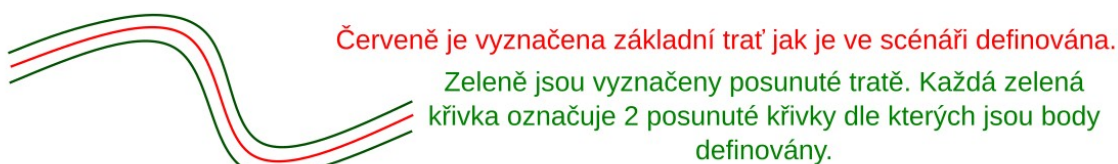
## 2.3. Vykreslování okolí

### 2.3.1. Vykreslování kolejí

#### Generace vertexových bodů tratí v simulátoru

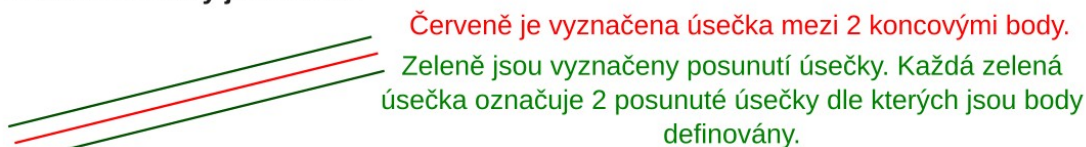
Šířka tratě, neboli rozestup mezi posunutými křivkami, je možné měnit.

Tratě definovány jako Bézierovy



Původní Bézierova křivka je kubická - definována 4 body: počátečním, konečným a 2 středními. Z každé křivky je vzat jeden bod. Ty jsou následně indexovány. Výška každého bodu je vypočítána z lineární interpolace dvou nejbližších výškových bodů.

Tratě definovány jako rovné



Z každé křivky je vzat jeden bod. Ty jsou následně indexovány. Pozice každého bodu je rovna lineární interpolaci koncových bodů s určitým poměrem. Rovná trať obsahuje stejný počet bodů jako je počet výškových bodů. Výška tratě v jednotlivém bodě je následně rovna výšce výškového bodu.

Obrázek 7: Obrázek popisující generaci tratí

Koleje jsou generovány za pomoci kubických Beziérových křivek dvourozměrném prostoru. Převýšení tratě se určuje za pomoci 10 tzv. výškových bodů uložených společně body potřebnými vygenerování křivky, kterých je možné nastavit hodnotu Y. Opravdová výšková hodnota bodů je tratí definovaných Bézierovými křivkami rovna lineární interpolaci výškových bodů.

Veškeré koleje jsou uloženy jednom páru vertexového indexového bufferu. Tato struktura má několik výhod – za prvé je použit pouze kreslicí příkaz, což drasticky zrychluje vykreslování, za druhé je vždy vidět všechny koleje, což značně pomáhá realismu.

### 2.3.2. Vykreslování silnic, země, ulic, kolejových svršků stěn

Pro texturové parcely, které jsou použity pro terén, povrch ulic, kolejové svršky i pro stěny, jsou při načítání scénáře generované vertexové indexové buffery. Veškeré texturové parcely jsou zkombinované do jednoho vertexového jednoho indexového bufferu, jehož body odkazují na jednotlivé materiály (a nimi spojené textury) systému GMS. Tento přístup, stejně jako tratí, má výhodu že je terén vždy vidět není ho potřeba při změně



zastávky měnit. Tento přístup má avšak také jednu nevýhodu, to ohraničení počtu různých typů texturovaných texturových parcel na 27 důvodů ohraničení počtu texturových slotů API OpenGL.

Jednotlivé textury byli staženy webového portálu AmbientCG.<sup>[22]</sup> Materiál na svrších či okolních silnicích nemá žádný vliv na fyzikální simulaci či pohyb vozidla.

### **2.3.3. Vykreslování zastávek a návěstidel**

Informační cedule informacemi linkách je nejdříve za pomoci framebufferu vykreslena do textury, která je později při instancovaném vykreslování staničních sloupů vykreslena. důvodů limitace OpenGL nemožnosti změny textury během jednoho (byť instancovaného) výkresového příkazu, jsou tyto textury vykreslovány zvlášť. Jízdní řády jednotlivých linek nejsou vykreslovány. Je simulováno také noční osvětlení sloupů, to za pomoci změn parametrů materiálů.

### **2.3.4. Vykreslování budov památek**

Vykreslování budov funguje za pomoci využití instancovaného vykreslování. Pro každý typ budov, kterých je simulátoru 4, je poslán grafické kartě pouze jeden příkaz na vykreslení objektů.

Jelikož památky jsou jedinečné a většinou jsou přítomny na mapě pouze jednou, není vykreslování památek instancováno, a z toho důvodu že instancování by nepřineslo žádné podstatné zrychlení programu.

## 3. Prvky předpisy ovládání simulátoru

### 3.1. Pohyb po trati návěstí

**Seznam tramvajových návěstí, jejich funkcí a vzhledu**  
 Návěsti jsou značky které určují co má řidič tramvaje dělat resp. nedělat.

<p><b>Základní návěsti</b></p> <div style="display: flex; align-items: center; margin-bottom: 5px;"> <p>Č. 1 označuje zajišťovací výhybky. Takovou je možné najít např. na obratišti Zvonařka.</p> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;"> <p>Č. 2 označuje návěst stůj. Vlak nesmí návěst projet. Používá se při výlukách.</p> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;"> <p>Č. 3 označuje přítomnost přejezdu s předností tramvaje.</p> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;"> <p>Č. 4 označuje maximální rychlost úseku je-li nižší než očekávána t.j. stanovena jinými předpisy.</p> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;"> <p>Č. 5 označuje konec úseku označeného značkou č. 4.</p> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;"> <p>Č. 6 označuje návěst "pomalu" neboli omezení rychlosti na 10 kilometrů za hodinu.</p> </div> <p><b>Rychlostní návěsti</b></p> <div style="display: flex; align-items: center; margin-bottom: 5px;"> <p>Č. 7 označuje úsek s maximální rychlostí.</p> </div>	<div style="display: flex; align-items: center; margin-bottom: 5px;"> <p>Č. 8 označuje konec úseku označeného značkou č. 7.</p> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;"> <p>Č. 9 označuje místo které tramvaj musí projet nejvýše uvedenou rychlostí.</p> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;"> <p>Č. 10 stanovuje rychlostní limit 10 km/h při odbočování z důvodu struktury výhybky.</p> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;"> <p>Č. 11 označuje oblouk o malém poloměru <math>r &lt; 25m</math> a maximální rychlost 15 km/h na něm.</p> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;"> <div style="display: flex; flex-direction: column; align-items: center;"> <div style="background-color: yellow; border: 1px solid black; padding: 2px;">ZONA 22-6 h</div> <div style="background-color: yellow; border: 1px solid black; padding: 2px;">30</div> </div> <div style="display: flex; flex-direction: column; align-items: center;"> <div style="background-color: yellow; border: 1px solid black; padding: 2px;">ZONA 22-6 h</div> <div style="background-color: yellow; border: 1px solid black; padding: 2px;">40</div> </div> </div> <p>Značky č. 12 a 13 určují maximální rychlost v určitý čas.</p> <div style="display: flex; align-items: center; margin-bottom: 5px;"> <div style="display: flex; flex-direction: column; align-items: center;"> <div style="background-color: yellow; border: 1px solid black; padding: 2px;">ZONA 22-6 h</div> <div style="background-color: yellow; border: 1px solid black; padding: 2px;">30</div> </div> <div style="display: flex; flex-direction: column; align-items: center;"> <div style="background-color: yellow; border: 1px solid black; padding: 2px;">ZONA 22-6 h</div> <div style="background-color: yellow; border: 1px solid black; padding: 2px;">40</div> </div> </div> <p>Značky č. 14 a 15 označují konec úseku určeného značkami 12 a 13.</p>	<p><b>Návěsti pro sběrač</b></p> <div style="display: flex; align-items: center; margin-bottom: 5px;"> <p>Č. 16 - Úsekový dělič. Tramvaj musí projet toto místo bez odběru proudu.</p> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;"> <p>Č. 17 - Úsekový dělič s magnety. Je možné toto místo projet s odběrem.</p> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;"> <p>Č. 18 - Stáhni sběrač. Označuje překážku v trolejovém vedení.</p> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;"> <p>Č. 19 - Zvedni sběrač. Po této návěsti je plně bezpečné zvednout sběrač.</p> </div>
--	--	---

Obrázek 8: Funkce, čísla vzhled jednotlivých tramvajových návěstí.<sup>[1][23][24]</sup>

Tramvaj se pohybuje po kolejkách. Rychlost akce tramvaje jsou dány návěstmi polohami zastávek.

Na obrázku nahoře je dán seznam často užívaných návěstí. Na rovných úsecích (tj. úsecích bez výhybek) je rychlost dána návěstmi č. až 12. Návěsti se mohou také týkat polohy sběrače – zda ho musí řidič tramvaje stáhnout či naopak zvednout. Tímto se zabývají návěsti č. 16 až 19.<sup>[1 \*]</sup>

Při uslyšení zvukové sirény vozidla IZS, musí řidič tramvaje neprodleně zastavit.<sup>[1 \*]</sup>

## **3.2. Výhybky**

### **3.2.1. Elektrické**

Elektrické výhybky jsou ovládány uzavřením elektrického obvodu výhybky za pomoci pantografu vozidla.

Elektrické výhybky je ke 22. březnu 2025 možné pražské tramvajové síti potkat pouze 2 místech, to na vjezdu do smyčky Dlabačov ze směru Pohořelce na lince 23 na vjezdu do smyčky Vysočanská. Do roku 2025 byla síti přítomna ještě jedna výhybka, to nedaleko zastávky Chotkovy sady, ta byla avšak přestavěna na rádiově ovládanou.

### **3.2.2. Rádiové**

Nejčastější typ výhybek na pražských tratích jsou rádiové. Rádiově ovládané výhybky mají rádiový přijímač, který čte informaci přehození výhybky poskytnutou tramvají. Dle přijetího signálu se přehodí do odpovídajícího směru.

### **3.2.3. Zajišťovací výhybky**

Zajišťovací výhybky, též označované jako vratné, jsou takové, které se po průjezdu vozu za pomoci pružiny vrátí na původní pozici. Takové výhybky se současně používají pouze na obratišti Zvonařka pro linku 23 ve vozovkách. Na seznamu návěstí (obrázek 8) jsou označeny číslem 1.

### **3.2.4. Přehazování výhybek**

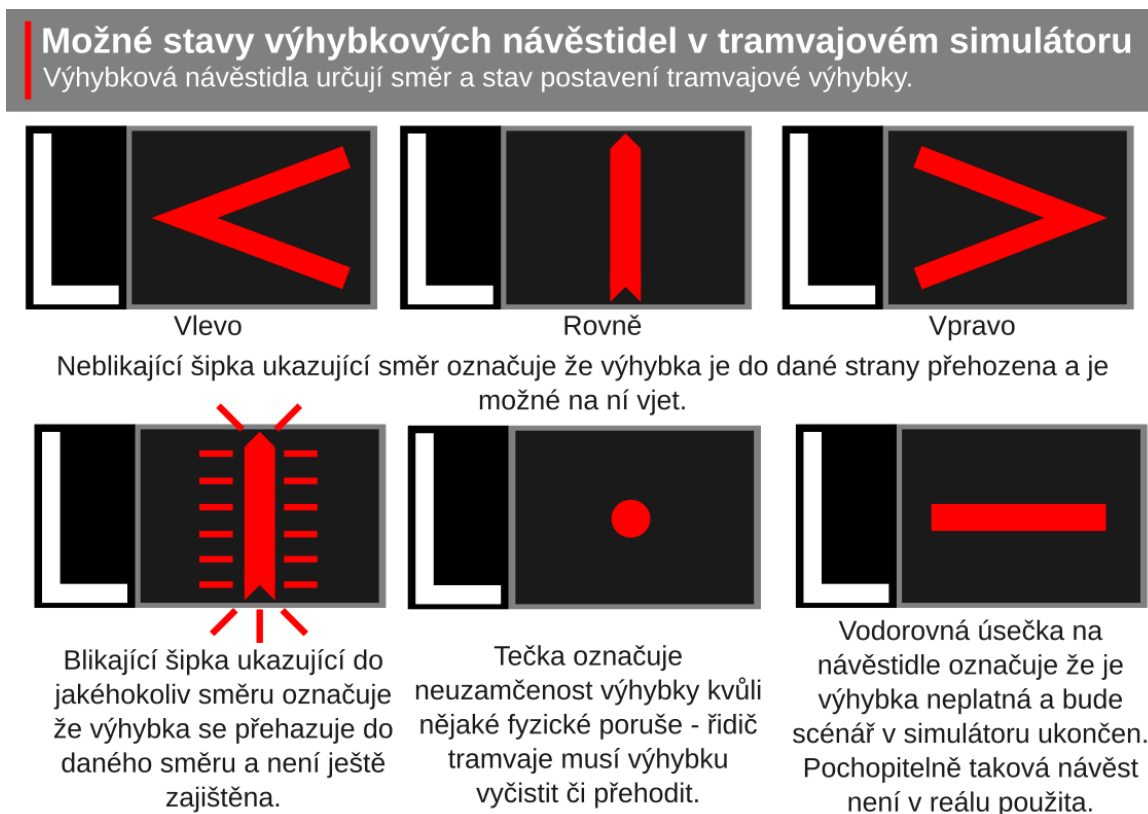
Přehazování výhybek závisí na typu výhybky. Elektrické výhybky jsou přehazovány trolejovým kontaktem. Rádiové výhybky jsou přehazovány

TODO CITACE KASTNER LEUBNER

Při poruše přehazování výhybky je zapotřebí přehodit danou výhybku ručně.

### 3.3. Návěstidla

#### 3.3.1. Výhybková návěstidla

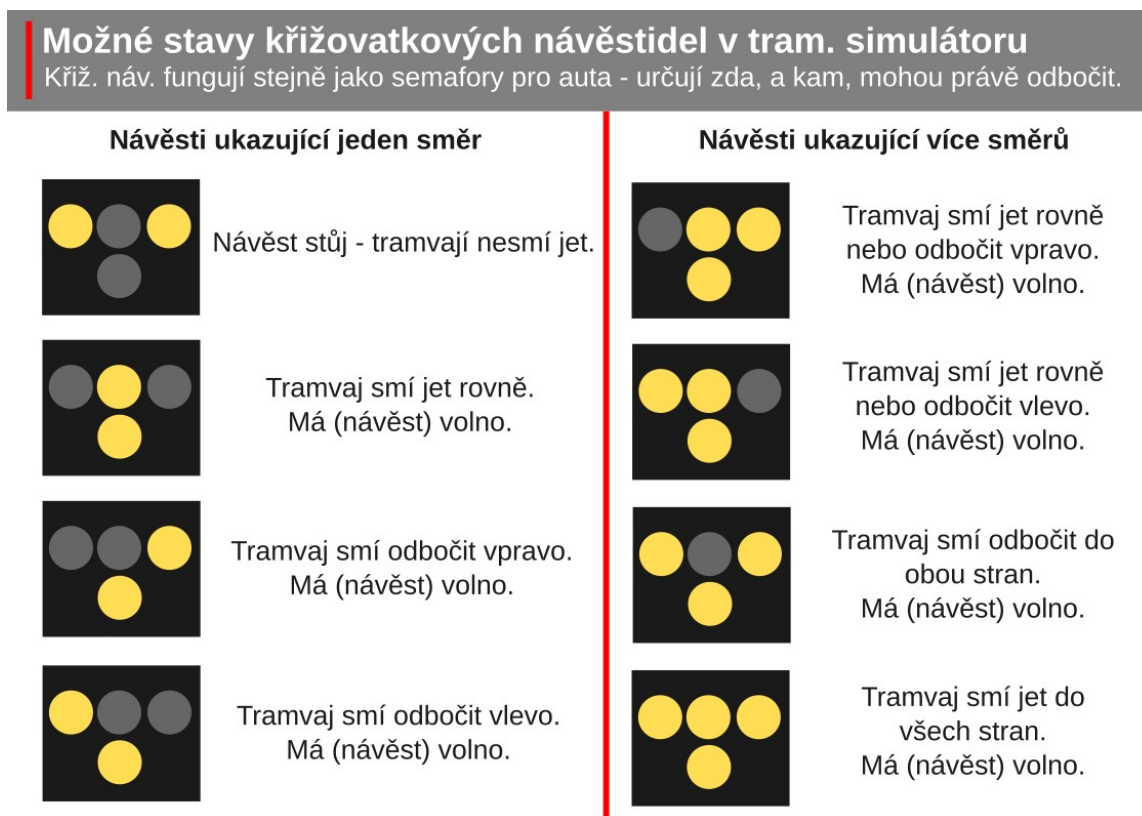


Obrázek 9: Možné stavy výhybkových návěstidel tramvajovém simulátoru.<sup>[1]</sup>

Návěstidla určují směr stav přehození výhybky. Na samotném displeji návěstidla je znázorněna červená šipka znázorňující směr do kterého je současně výhybka přehozena. Blikající návěst označuje že výhybka není ještě zajištěna je pouze procesu přehazování.<sup>[1\*]</sup>

Tramvaj nesmí opustit sféru vlivu návěstidla dokud není výhybka přehozena zajištěna. Vyjede-li tramvaje ze sféry vliv návěstidla, scénář se ukončí.

### 3.3.2. Křižovatková návěstidla



Obrázek 10: Možné stavy křižovatkových návěstidel tramvajovém simulátoru.<sup>[1][25]</sup>

Křižovatková návěstidla plní funkci semaforu pro tramvaje. Jednotlivé funkce křižovatkových návěstidel jsou obsaženy obrázku.

Simulátor současně nesimuluje výzvoová návěstidla, to důvodů jejich nečetnosti jejich nevelkého vlivu na dopravní provoz. Na rozdíl od „normálních“ křižovatkových výhybkových návěstidel, která jsou kontrolována odpovědnými drážními předpisy, jsou návěstidla výzvoová kontrolována napřímo Dopravním podnikem hl.m. Prahy, tak jsou užity pouze místech kde se na křižovatce míjí pouze tramvaje. na takových křižovatkách jsou avšak vždy doprovázeny čočkovými návěstidly.<sup>[1\*]</sup>

### 3.3.3. Předvěsti křižovatkových návěstidel

Křižovatková návěstidla mohou mít také předvěsti, která ukazují řidiči stav následujícího návěstidla. Od obvyklých křižovatkových návěstí se vizuálně liší pouze žlutou tabulkou černým písmenem “P”. simulátoru se od obvyklého také liší tím, že nemá přiřazenou žádnou sféru vlivu či žádnou funkčnost – pouze informuje stavu následujícího návěstidla.<sup>[25\*]</sup>

## **3.4. Jízdní řád**

### **3.4.1. Normální zastávky**

Na normálních zastávkách je vůz povinen zastavit. Není-li zastávka kontrolním bodem, není dán přesný čas kolik musí řidič na zastávce strávit. Není-li na zastávce žádný cestující přítomen, nemá řidič za povinnost otevírat dveře.

### **3.4.2 Zastávky na znamení**

Na zastávkách na znamení nemusí řidič vždy povinně zastavit. Je-li na zastávce cestující, zastavit je avšak povinen. Nemá-li řidič jistotu přítomnosti cestujících na zastávce, musí taktéž zastavit.

### **3.4.3 Kontrolní body**

Stejně jak je to reálných jízdních řádů, řidiči musí dodržovat časy pouze pro některé zastávky, tzv. kontrolní body. Kontrolní body jsou definovány pro každou linku zvlášť. kontrolních bodů není možné odjet předčasně.

Ve vnitropodnikových jízdních řádech vozidla jsou specifikovány pouze časy kontrolních bodů, nikoliv všech zastávek.

### **3.4.4. Konečné zastávky**

Na konečných zastávkách končí linky svou jízdu. Toto však neznamená že řidič končí směnu nebo že řidič vždy končí směnu na konečné.

V simulátoru nejsou přidány žádná ohraničení na konečné zastávky. Toto řešení povoluje simulovat části linek či linky výlukové, které nemusí povinně končit na obratiště, nýbrž taktéž na úvratí.

CITACE KASTNER LEUBNER KAŽDÝ PARAGRAF 3.4.1. – 3.4.4.

### 3.5. Identifikace tramvaje

Tramvaj je identifikována dvěma hodnotami – číslem vozovny, pořadí výjezdu vozovny. Pro časový rozsah není pro jejich tvorbu žádné jednoznačné pravidlo. To také znamená, že pořadí není určeno dle času odjezdu.<sup>[26\*]</sup>

Tyto hodnoty jsou závislosti na typu vozu tramvaje zobrazeny na různých místech. Novější tramvaje typu KT8D5.RN2P, 14T, 15T rekonstruované vozy T3R.PLF mají pořadí zobrazené digitálně (kromě T3R.PLF) na straně vozu. Vozy typů T3R.P T3R.PLF mají své pořadí uvedené na čelní straně vozu kabině řidiče to ve formě plastové destičky.<sup>[1]</sup>

Název vozovny	Kód vozovny
Vozovna Hloubětín	1
Vozovna Kobylisy	2
Vozovna Motol	3
Vozovna Pankrác	4
Vozovna Strašnice	5
Vozovna Střešovice	6
Vozovna Vokovice	7
Vozovna Žižkov	8

Tabulka 5: První číslice dle vozovny<sup>[1][26]</sup>

První číslice pořadového čísla určuje vozovnu. Přiřazení čísel jednotlivým vozovnám je uvedeno tabulce. Druhá třetí číslice určuje pořadí ve vozovně. Toto číslo závisí na jednotlivé vozovně na na času odjezdu není ho možné přesně simulovat.

Pořadové číslo spoje není to stejné jako pořadí na lince. Pořadí na lince, které je dostupné cestujícím on-line, označuje pořadí spoje rámci určité linky. Obvykle je uvedeno ve formátu *LL/P*, kde označuje linku *P* označuje pořadí vlaku na lince. Pořadové číslo není na veřejných zdrojích uvedeno jedná se jednoduché číslo. Pořadové číslo linky se čte následovně:

***P-tá LL-ka: např. devátá šestnáctka***

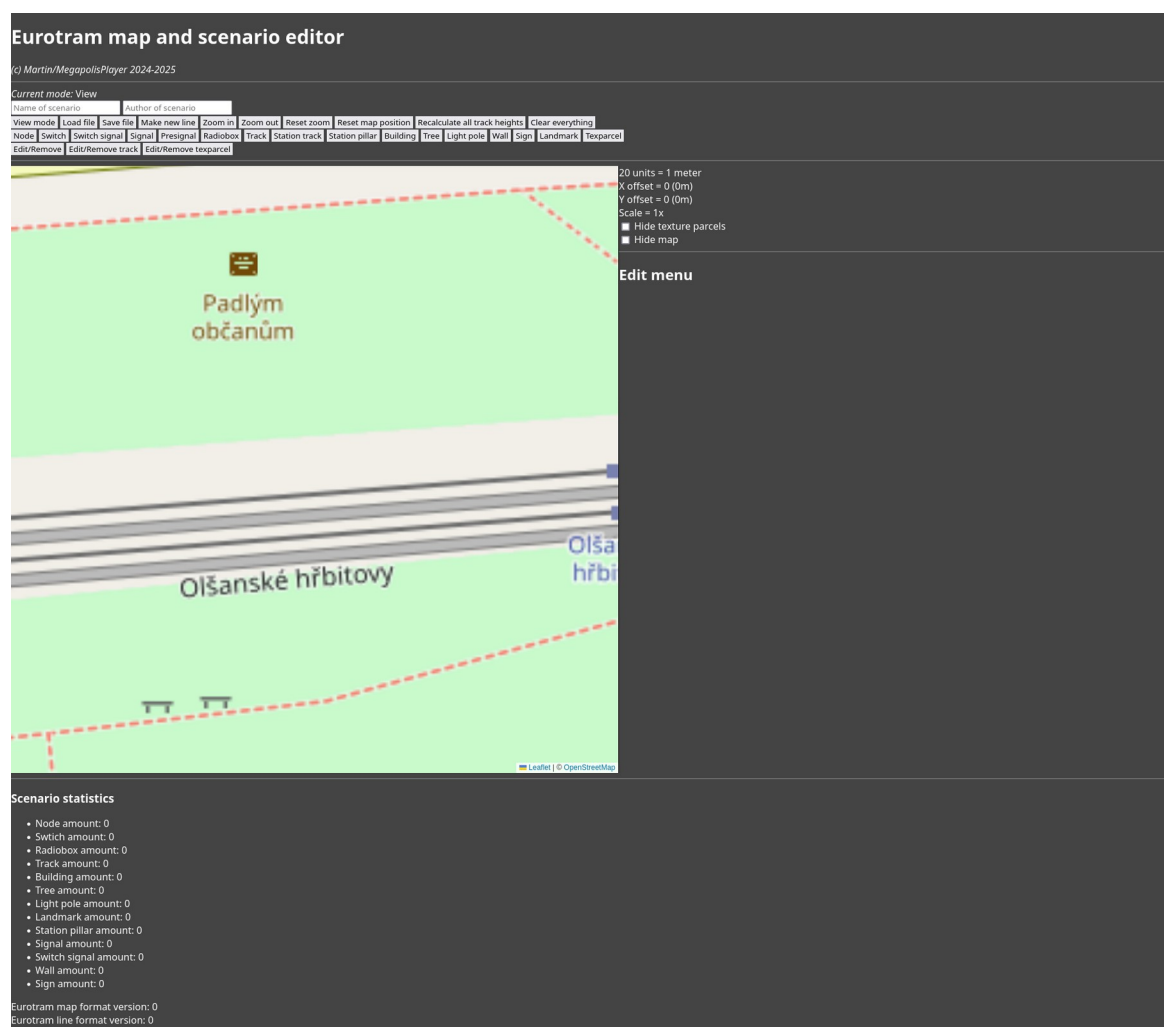
[1\*][26\*]

## 4. Dodatečné nástroje formáty jejich souborů

### 4.1. Nástroj na úpravu mapy scénářů

#### 4.1.1. Základní informace

Nástroj na úpravy (jinak označovaný jako „editor“ map scénářů) je webová aplikace napsaná programovacím jazyce JavaScript (za pomoci dodatečných technologií HTML CSS).<sup>[27]</sup> Funguje jako zcela nezávislý program který exportuje „mapu“ (kolekci tratí, zastávek okolí) do vlastního formátu. Exportuje také „scénáře“, neboli soubory informací jednotlivých hrách ve kterém je uložena linka (včetně případného přejezdu na jinou či zatahování do depa) její jízdní řád.



Obrázek 11: Fotografie nástroje pro úpravu mapy scénářů při jeho otevření.



Editor je rozdělen na části – horní lištu operací, menu úprav, plátno statistiku.

Statistika pouze ukazuje informaci scénáři. Ukazuje verzi formátu souboru kterou editor vytváří počet prvků ve všech seznamech objektů scénáří.

Úpravy probíhají na mapovém podkladu, vytvořeným za pomoci technologie Leaflet.js<sup>[28]</sup> používající mapu OpenStreetMap.

Ukládání probíhá ve dvou stupních. Nejdříve je veškerá informace uložena do pole bytů. Toto pole bytů je následně převedeno na soubor staženo. Načítání funguje opačně – nejdříve se načte souboru do pole bytu, pak se byty za sebou přečtou interpretují.

Editor sice desetinná čísla nezakazuje, snaží se je ale omezit. Při ukládání do souboru se všechny hodnoty převedou na jejich dolní celá čísla.

#### **4.1.2. Horní lišta**

Horní lišta operací ukazuje seznam akcí které může uživatel provést. této lišty je možné přidat body, výhybky, návěstidla, rádiové čtečky, tratě, staniční tratě, stromy, pouliční lampy, budovy, památky, stěny parcely.

Body určují kudy trať prochází. Znají svojí polohu, výšku, sounáležitost ke stanici. Bod může také hraničit na pomezí stanic – takovém případě je při načtení souboru simulátorem bod duplikován.

Výhybky plní stejnou funkci jako body. Znají avšak toho po jakém bodě následují, body na které do každé strany směřují k ní připojenou rádiovou čtečku návěstidlo. Výhybky nemohou hraničit jinou stanicí.

V programu je možné najít typy návěstidel – výhybková křižovatková. Výhybkové návěstidlo ukazuje stav směr výhybky. Křižovatkové návěstidlo funguje jako alternativa semaforu. Fungování návěstidel je podrobněji popsáno kapitole 3.3 „Návěstidla“.

Rádiové čtečky čtou ukládají požadavek tramvaje, který je pak přečten výhybkou která se následně odpovědně přehodí. Stav výhybky je následně vypsán na výhybkové návěstidlo.

Tratě jsou uloženy jako Beziérovky křivky nebo přímky mezi body. Obsahují také 10 rovnoměrně rozložených „výškových bodů“ určujících změnu převýšení. Existuje také tzv. „staniční trať“ která určuje místo zastavení tramvaje. Staniční tratě normální tratě jsou uloženy ve stejném poli.

Stromy ukládají pouze svojí polohu výšku. Pouliční lampy ukládají také svůj úhel stupních.

Budovy ukládají kromě své polohy výšky také svůj typ. Rozměr budovy není ukládán jelikož závisí na typu budovy může se změnit pouze změnou-li se také typ budovy. Jak nástroji na plánování, tak v simulátoru je uložena tabulka rozměry odpovídajícími typu budovy.

Památky jsou zvláštní typ budov. Ukládají pouze polohu, výšku 4-písmenný kód který je pak simulátoru propojen modelem dané památky či neobvyklé budovy.

Stěny ukládají pouze svoje body, svojí výšku název svého materiálu.

V každém objektu (kromě památek) je také uložena kód stanice které je objekt součástí. Kód stanice je uložen jako 4-písmenný kód důvodů optimalizace. simulátoru je tento kód uložen jako 32-bitové číslo. toho důvodu je podporována pouze písmenná sada UTF-8 ve které každé písmeno je uloženo jako byte. simulátoru jsou vždy vykreslovány pouze některé stanice, to ty mezi kterými se tramvaj současně nachází stanice po tom následující. Je-li tomto úseku výhybka, jsou načteny stanice ze všech stran výhybky.

Pro modelování silnic uličního povrchu je možné přidat tzv. „parcely“, které vymezují určitý prostor určité výšce. Každá parcela je čtyřúhelník který kromě polohy výšky všech svých bodů zná také jakým dvěma stanicím náleží název materiálu. Simulátor pak dle uloženého názvu najde odpovědný materiál vykreslí nim parcelu.

### **4.1.3. Menu úprav**

Menu úprav je umístěno pravé strany od plátna. tomto menu je možné měnit přiblížení plátna, nastavovat některé parametry vykreslování plátna upravovat jednotlivé objekty. Jsou dostupné parametry vykreslování plátna to vypnutí vykreslování parcel vypnutí mapového podkladu.

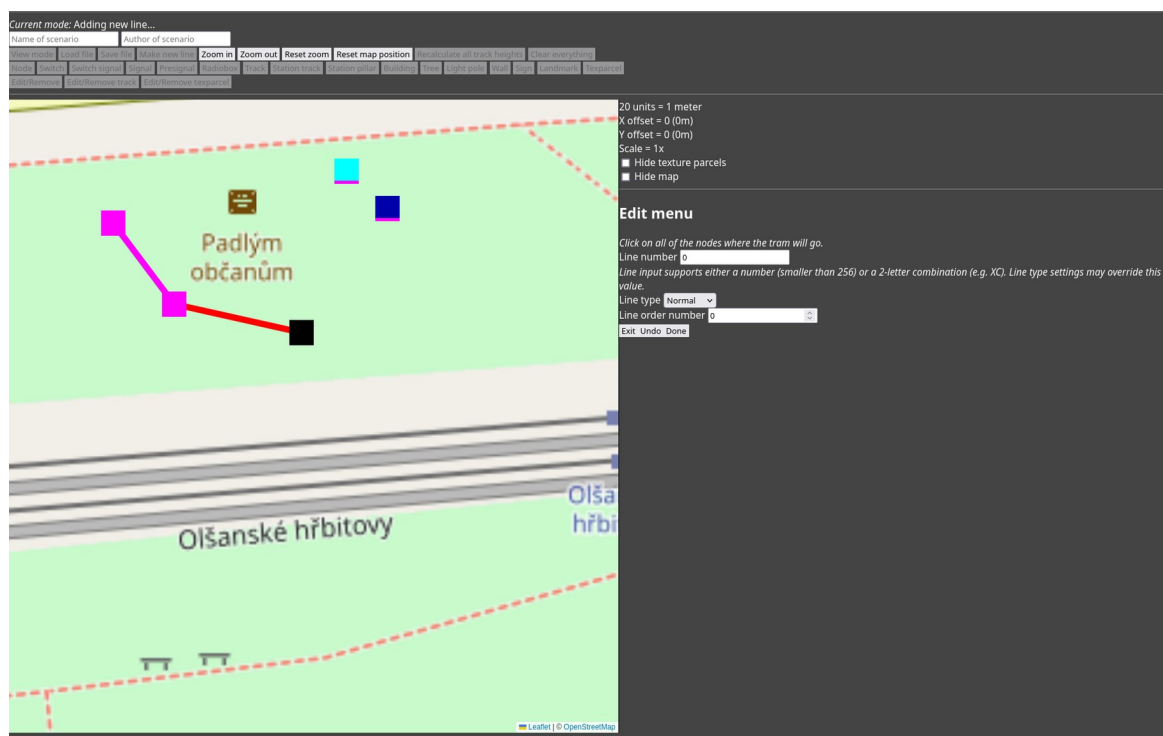
Menu úprav je dynamicky generováno pro každý objekt. Před generací je menu resetováno.

### **4.1.4. Plátno úprav**

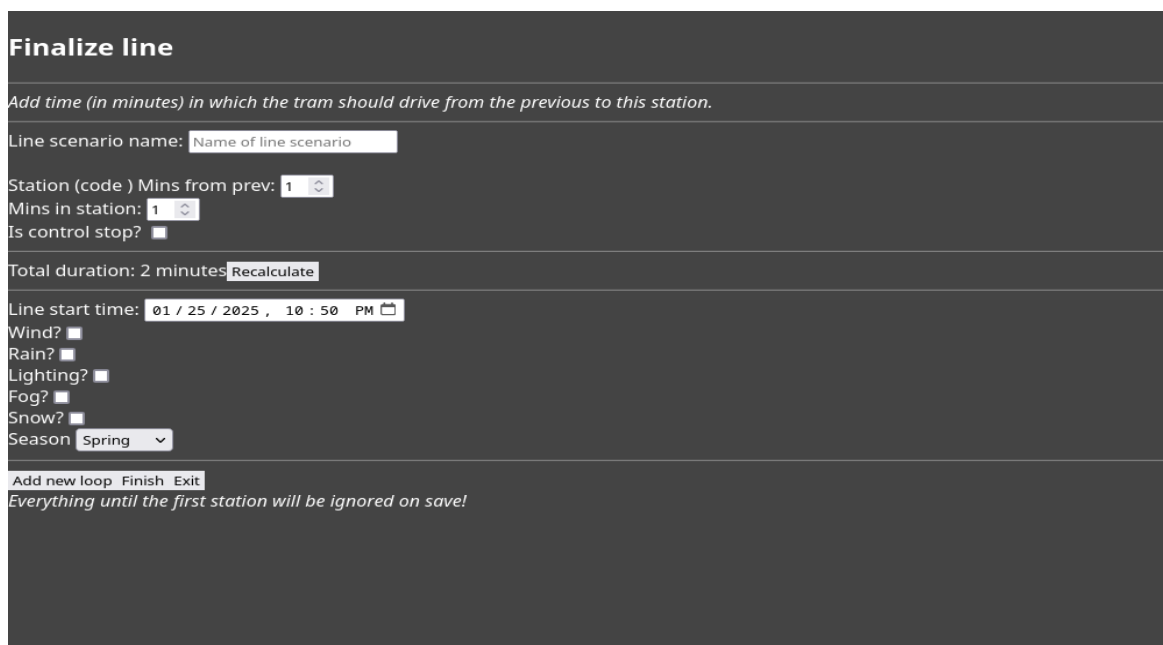
Plátno editoru je možné zmenšovat či zvětšovat pohybovat ním. Toto, vykreslování objektů jejich správných místech je docíleno použitím transformačních matic. Při vykreslování objektu je nejdříve aplikována transformace do centra objektu. Následně, kdy centrum objektu je počátkem soustavy, je aplikována rotace kdy je matice otočena odpovídající počet stupňů, uvedený objektu. Následně je aplikována translace polovinu rozměru objektu, to důvodu přenesení počátku na levý horní roh objektu, což všechny kreslicí funkce očekávají.

Objekty, které jsou za hranicí plátna nejsou vykreslovány. Samotné plátno je také naprosto průhledné jelikož obsahuje také mapový podklad mapy OSM za pomoci technologie Leaflet.js. Pod mapou plátnem je také uložené bílé pozadí, které je vidět při vypnutí mapy. Mapu je možné během upravování vypnout. Na základní mapový podklad vytvořený technologií Leaflet.js bylo napojeno vlastní ovládání, které pohybuje nejen plátnem ale také mapou.<sup>[28 \*]</sup>

## 4.1.5. Tvorba linky



Obrázek 12: Fotografie režimu vytváření linek nástroji pro úpravu mapy scénářů.



Obrázek 13: Fotografie menu nastavení linek nástroji pro úpravu mapy scénářů.

Tvorba linky s ní spojeného souboru scénáře probíhá po stisknutí tlačítka vytvoření nové linky. tomto režimu se nejdříve vybírá trasa prvního kola linky klikáním na body výhybky,

po čemž se nástroj přepne to menu nastavení linky. tomto menu může uživatel buď linku ukončit, po čemž se všechna kola linky uloží do souboru nebo přidat nové kolo linky, čímž se cyklus uzavře. Během přidávání kola či úpravy kol existujících je možné režim opustit bez ukládání linky. tomto režimu úprav se také přidává jízdní řád, to ve formě ukládání času jízdy mezi zastávkami minutách. Režim přidávání linek, to ve stavu přidávání tratí do určitého kola, je znázorněn na následujícím obrázku.

## 4.2. Formát souboru okolí

Soubor okolí má příponu .etmap („Eurotram map“) je sestaven hlavičky, udávající základní informace seznamů objektů. Hlavička má následující formát:

ETMP D I Y R L B W X J . . . J0 A . . . A0 E

Všechny soubory tohoto formátu musí začínat byty odpovídajícími textu „ETMP“ kódování ASCII. Dále následuje 16-bitové číslo označující verzi formátu číslem od 0 do 65535 ( $2^{16} - 1$ ). Program musí povinně ukázat chybu je-li číslo verze odlišné od verze popřípadě verzí kterou může program přečíst. Je 64-bitové číslo označující unixový čas milisekundách při poslední změně souboru. Číslo  $N$ ,  $I$ ,  $T$ ,  $Y$ ,  $S$ ,  $R$ ,  $P$ ,  $L$ ,  $G$ ,  $B$ ,  $M$ ,  $W$ , a jsou 32-bitová určují počet jednotlivých typů objektů ( $N$  určuje počet bodů, počet výhybek, tratí staničních tratí, výhybkových návěstidel, křižovatkových návěstidel předzvěstí, rádiových přijímačů, staničních sloupů, pouličních lamp, stromů, budov, památek, stěn, značek  $X$  parcel) scénáři. a jsou textové řetězce libovolné délce označující jméno scénáře autora mapy. Oba řetězce jsou zakončeny nulovým znakem. 8-bitové číslo určuje počet pixelů na metr.

Seznam objektů je za sebou uspořádaný dle typů. rámci typů nejsou avšak nijak tříděny. Poloha (označené následujících ukázkách jako čísla  $X$ ,  $Y$ ) výška (označená následujících ukázkách jako  $H$ ) jsou vždy uloženy jako 32-bitová čísla. Rotace (označená následujících ukázkách jako  $R$ ) je vždy uložena ve stupních je uložena jako 16-bitové číslo. Staniční kódy (označené následujících ukázkách jako  $SSSS$ ) jsou vždy uloženy jako 4-písmenné řetězce uloženy ve formátu ASCII které je možné jednoduše převést na interpretovat jako 32-bitová čísla. Staniční kód by měl být pro každou stanicí unikátní. Stanici několika stanovišti je přiřazen jeden kód. Nemá-li objekt přiřazený staniční kód, je vykreslován. Památky jsou vždy vykreslovány.

Body jsou uloženy následujícím formátu:  $Y$   $SSSS$   $BBBB$

Nejen  $SSSS$ , ale také  $BBBB$  jsou staniční kódy označující stanici ke kterému bod náleží stanici se kterou bod hraničí. Pokud bod nehraničí se žádnou jinou stanicí, jsou všechny byty nulové.

Výhybky jsou uloženy následujícím formátu:  $Y$   $SSSS$   $L$   $P$   $G$   $T$

32-bitové číslo určuje pořadí bodu který slouží jako trať směřující výhybce. Směry odboček jsou považovány ze směru bodu pořadí  $B$ . 32-bitové číslo označuje bod do kterého vede levá odbočka výhybky. 32-bitové číslo označuje bod do kterého vede výhybka, je-li nastavena rovně. 32-bitové číslo označuje bod do kterého odbočuje prvá odbočka výhybky. Číslo  $B$ ,  $L$ , a se nesmí rovnat pořadí výhybky. Pokud je číslo  $L$ , a rovno -1, výhybka do daného směru neodbočuje. Jsou-li  $L$ , i rovny -1, je výhybka neplatná při jejím průjezdu ukončen současný scénář. Je-li číslo rovné -1, je výhybka také neplatná. 32-bitové číslo určuje pořadí rádiového přijímače výhybky. 32-bitové číslo určuje pořadí návěstidla výhybky. Písmeno  $T$ , uložené kódování UTF-8, označuje písmeno návěstidle označující tuto výhybku.

Tratě jsou uloženy následovně: TK N A C H1 . . . 9

První byty odpovídají textu „TK“ kódování ASCII. 32-bitová čísla a určují pořadí prvního druhého bodu které trať spojuje. 8-bitové číslo určuje flagy. objektu jsou uloženy flagy – zda je první bod výhybka či ne, zda je druhý bod výhybka či ne zda je trať přímka či Beziérova křivka. Flagy jsou uloženy uvedeném pořadí. 32-bitová čísla A, určují polohu prvního kontrolního bodu Beziérovky křivky, zatímco 32-bitová čísla C, určují polohu druhého bodu. Pokud je trať přímka, jsou tyto hodnoty nastavené na nulu. Čísla H1 až H9 určují deset výškových bodů tratě, rovnoměrně rozprostřených podél délky tratě. Jelikož změna výšky může být také desetinným, jsou výškové body stonásobkem jejich opravdových hodnot. Tratě nesmí vést mezi výhybkami. Každý bod který není výhybkou smí mít maximálně tratě. Všechna čísla jsou 32-bitová.

Staniční tratě jsou uloženy následovně: ST N A C H1 . . . 9 SSSS

První byty odpovídají textu „ST“ kódování ASCII. Staniční tratě ukládají stejnou informaci jako tratě svůj staniční kód, který určuje jakou stanici se jedná. Tratě staniční tratě se rozlišují na základě prvních bytů. Tratě staniční tratě jsou uloženy společně.

Výhybková návěstidla jsou uloženy následovně: Y R SSSS L . . . L

8-bitové číslo určuje počet jednotek návěstidel. Každá jednotka je identifikována svým písmenem L. Každé písmeno je uloženo jako 8-bitové číslo kódování UTF-8.

Křižovatková návěstidla jsou uloženy následovně: JS Y R SSSS

První byty odpovídají textu „JS“ kódování ASCII. je 16-bitové číslo označující výšku sloupu metrech. Je-li rovno nule, je křižovatkové návěstidlo vykreslováno pod trakčním vedením, na stejné úrovni jako značky. Výška ve kterém je vykreslováno návěstidlo se rovná součtu a P.

Předvěsti pro křižovatková návěstidla jsou uloženy následovně: PJ Y R I SSSS

První byty odpovídají textu „PJ“ kódování ASCII. Stejně jako křižovatkového návěstidla, je 16-bitové číslo označující výšku sloupu metrech. 32-bitové číslo označuje pořadí návěstidla ze kterého předvěst čerpá informaci. Předvěsti křižovatková návěstidla jsou uloženy společně.

Rádiové přijímače stromy jsou uloženy následovně: Y SSSS

Staniční kód radiového přijímače musí odpovídat staničnímu kódu jeho výhybky.

Staniční sloupky pouliční lampy jsou uloženy následovně: Y R SSSS

Budovy jsou uloženy následovně: Y R SSSS

Číslo typu	Název typu	Popis	Šířka, hloubka
0	Normální	Standardní městská budova, typická pro centrum.	20m, 19m
1	Rohová	Rohová městská budova, typická pro centrum.	24m, 24m
2	Rodinná	Rodinná vila, typická pro Hanspaulku.	20m, 12m
3	Panelová	Vysoká panelová budova, typická pro sídliště.	25m, 15m

Tabulka 6: Seznam typů budov, jejich popisků rozměrů.

Číslo je 8-bitové určuje typ budovy dle následující tabulky.

Památky jsou uloženy následovně: Y R T

Číslo je památek 16-bitové určuje typ budovy dle následující tabulky. simulátoru jsou obsaženy pouze ty památky, které jsou podél tratě vidět, nikoliv všechny které je možné Praze nalézt.

Kód památky	Název památky	Typ budovy
OLHB	Olšanské hřbitovy	Administrativní
ATRF	Atrium Flora	Obchodní centrum
JZPC	Kostel na nám. Jiřího Poděbrad	Kostel
NTM1	Národní muzeum, stará budova	Muzeum
NTM2	Národní muzeum, nová budova	Muzeum
NMMC	Kostel na nám. Míru (sv. Ludmila)	Kostel
VRSC	Zámeček na Vršovickém nám.	Historická
CCHC	Kostel na Čechovo nám.	Kostel
ZZTV	Žižkovská televizní věž	Historická

Tabulka 7: Seznam památek, jejich popisků rozměrů.

Památky neukládají sounáležitost ke stanici jelikož jsou vždy vykreslovány.

Stěny jsou uloženy následovně: B C J SSSS M . . M0

Číslo A, a C, určují polohu obou bodů stěny. 16-bitové číslo určuje výšku stěny. je textový řetězec určující název materiálu stěny.

Značky, odborně návěsti, jsou souboru uloženy následovně: Y R SSSS T

Kde je 16-bitové číslo určující typ návěsti. Seznam všech návěstí, jejich funkcí, čísel vzhledu je možné najít kapitole „Pohyb po trati návěsti“.

Parcely jsou uloženy následovně: B D F H J L SSSS TTTT M...M0

32-bitová čísla až jsou hodnoty jednotlivých bodů. Hodnoty jsou uskupeny podle náležitosti bodu (tj.  $X_1, Y_1, H_1 \dots X_4 Y_4 H_4$ ) určují polohu výšku všech čtyř vrcholů parcely. Jsou také uloženy staniční kódy ( $S\_T$ ) název materiálu parcely  $M$ . Dle názvu materiálu najde simulátor ve svém seznamu odpovídající materiál vykreslí parcelu ním. Body jsou uloženy po směru hodinových ručiček.

Všechna čísla jsou se znaménkem podporují záporné hodnoty za pomocí dvojkového doplňku. Všechny hodnoty jsou uloženy pixelech jsou převedeny na metry za pomocí hodnoty uložené hlavičce souboru. Mezery jsou čistě grafickým dodatkem nejsou ve formátu (mimo řetězce) nikdy obsaženy. Čísla jsou uloženy ve formátu little-endian.



### 4.3. Formát souboru scénáře

Soubor scénáře má příponu .etscr („Eurotram scenario“) je sestaven hlavičky jízdního řádu.

Hlavička souboru má následující formát:

ETSC D W J . . . J0 A . . . A0

Všechny soubory tohoto formátu musí začínat byty odpovídajícími textu „ETSC“ kódování ASCII. 64-bitové číslo určuje unixový čas. 64-bitové číslo určuje počáteční čas scénáře unixovém čase. udává čas herní, nikoliv reálný. Simulátor nemusí hodnotu dodržovat, může si nastavit vlastní či změnit zadanou. je 16-bitové bitové pole ukládající flagy určující počasí během scénáře. také obsahuje informaci ročním období ve kterém se scénář odehrává. Současně není změna počasí během scénáře podporována. Roční období časy východu či západu slunce jsou vypočítány na základě informací obsažených S. Textový řetězec označuje jméno scénáře. Textový řetězec označuje jméno autora. Oba řetězce jsou zakončeny nulovým terminátorem (bytem hodnotě nula následujícím ihned po konci řetězce). Při odstranění jakéhokoliv bodu na mapě nástroji pro úpravu map scénářů jsou všechny scénáře zneplatněny musí být vytvořeny znova, to důvodů změn pořadí bodů. Přidávání nových bodů či jakékoliv operace objekty kromě bodů, výhybek tratí nemají na soubory scénáře vliv.

N je 8-bitové číslo označující počet kol od konečné do konečné které jsou scénáři obsažené. Každé kolo má hlavičku, seznam výhybek seznam zastávek. hlavička je obsaženo číslo linky, čas odjezdu konečné typ jízdy – jedná-li se jízdu služební, výjezd z/zájezd do vozovny či normální kolo na lince. seznamu zastávek výhybek jsou obsaženy čísla směry výhybek, kódy zastávek jízdní řád zastávek jejichž územím tramvaj projíždí.

Hlavička kola má následující formát: M N T

L je 8-bitové číslo označující číslo linky. je 8-bitové písmeno kódování UTF-8 určující druhé písmeno linky. Je-li linka číselná, je tato hodnota nastavena na a je použita pouze hodnota L. Je-li linka označena písmeny (např. XA, XC, X5 apod.) je tato hodnota nenulová L je považováno jako první písmeno lince kódované ve formátu UTF-8. Je-li nastaveno na jinou hodnotu, je linka považována manipulační ( $M = 1$ ), služební ( $M = 2$ ), zvláštní ( $M = 3$ ), cvičnou ( $M = 4$ ) či zkušební ( $M = 5$ ) hodnota je ignorována. závislosti od typů linky jsou jinak vykreslovány panely. je 16-bitové číslo označující pořadové číslo spoje (formát pořadového čísla je detailněji popsáno kapitole 3.5 „Identifikace tramvaje“). 16-bitové číslo označuje počet stanic pro dané kolo. 32-bitové číslo označuje pořadí prvního bodu kterým vozidlo projíždí. je písmeno kódování UTF-8 které je nastaveno na ,N' je-li první bod standardní bod nebo na ,S' je-li první bod výhybkou.

Seznam zastávek výhybek je zamíchaný. Uloženy jsou nejen standardní zastávky, ale také služební zastávky na obratištích či ve vozovnách. Pro každou zastávku je uložen počet výhybek které tramvaj musí zajistit správný směr (nikoliv pouze projet) cestě do další zastávky. je uloženo jako 8-bitové číslo. První položka je vždy zastávka. Zastávky jsou tím pádem uloženy následovně: I CCCC M N

První byte odpovídá písmenu „S“ kódování ASCII. je 32-bitové číslo určující pořadí staniční tratě na mapě. je staniční kód zastávky kódování ASCII. je 16-bitové číslo určující čas od minulé zastávky, to minutách. je 16-bitové číslo označující délku postoje tramvaje na zastávce, to minutách. Je-li postoj tramvaje na zastávce delší než minuty, může uživatel simulátoru ten čas přeskočit nejpozději do minuty před odjezdem. je 8-bitové číslo, které je rovno jedné pouze pokud je zastávka kontrolním bodem. Není-li zastávka kontrolním bodem, je rovno nule.

Výhybky jsou uloženy následovně: I D

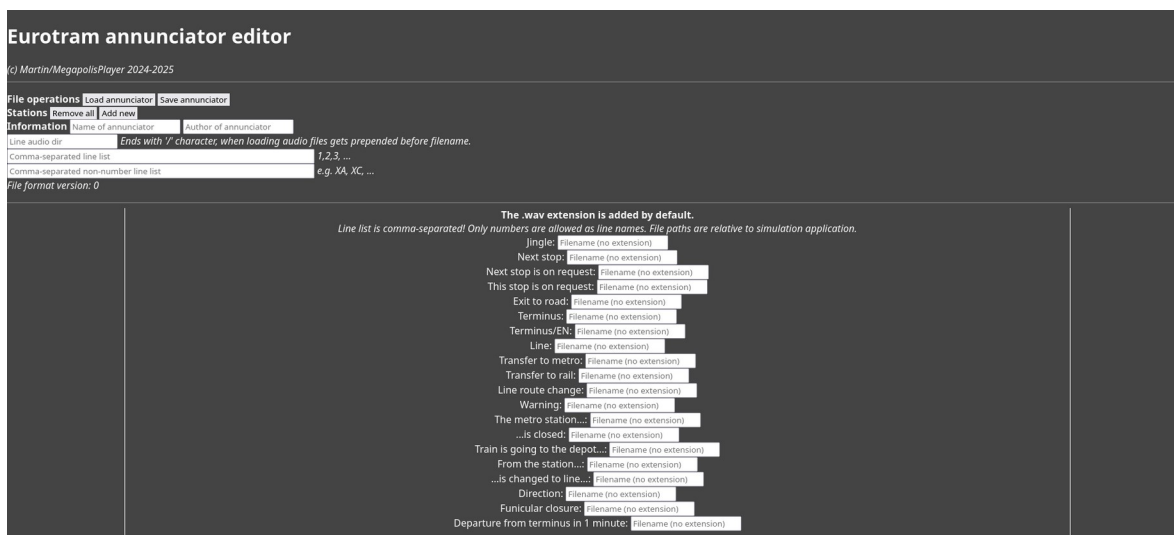
První byte odpovídá písmenu „W“ kódování ASCII. je 32-bitové číslo pořadí výhybky na mapě, zatímco 8-bitové číslo určuje směr přehození výhybky. Je-li = 1, je výhybka přehozena vlevo, je-li = rovně nebo je-li = vpravo. Je-li = 3, najíždí se dané situaci proti hrotu výhybky není potřeba jí přehazovat jelikož se přehodí sama. Je-li rovno jiné hodnotě, výhybka nemá žádný směr při jejím průjezdu je scénář ukončen. Začíná-li linka na úseku trati hned za výhybkou, není potřeba výhybku přehazovat.

Normální body tratě nejsou uloženy. Výhybky simulátoru, stejně jako výhybky na opravdových tratích, zjišťují směr tramvaje pouze když tramvaj již projíždí okolo rádiového přijímače výhybky. Je-li mezi stanicemi několik výhybek, začne tramvaj vysílat jiný signál po přejezdu výhybky za pomoci dodatečného čidla.<sup>[1]</sup>

Všechna čísla jsou se znaménkem podporují záporné hodnoty za pomoci dvojkového doplňku. Mezery jsou čistě grafickým dodatkem nejsou ve formátu (mimo řetězce) nikdy obsaženy. Čísla jsou uloženy ve formátu little-endian.

## 4.4. Nástroj na vytváření hlásiče

Nástroj na vytváření hlásiče je, stejně jako ostatní nástroje, webovou aplikací napsaná programovacím jazyce JavaScript (za pomoci technologií HTML CSS). Nástroj exportuje soubor hlásiče, který je pak simulátorem načten použit. Nástroj neslouží nahrání zvukové informace, nýbrž odkazování na složku která zvukovou informaci ukládá.



Obrázek 14: Fotografie nástroje pro vytváření hlásiče při spuštění

V nástroji je možné nastavit stav jednotlivých zastávek – zda jsou přítomny, uzavřeny či omezeny přestupní možnosti na metro nebo na železnici, jaké linky zastávkou projíždí jaký zvukový soubor obsahuje její název. Soubory hlásiče jsou rozděleny na dvě části – základové, neboli ty které nesouvisí ze zastávkami zastávkové, které jsou pro každou zastávku různé. kategorii zastávkových jsou uloženy jednotlivé názvy zastávek. kategorii nesouvisejících jsou uloženy obecná hlášení, například hlášení pro příští zastávku nebo pro konečnou zastávku. Nástroj umí soubory hlásiče jak vytvářet ukládat, tak načítat upravovat. nástroji je také možné nastavit jaké linky mají hlášení přítomné simulátoru.

Zvuky jednotlivých vozů jsou uloženy jejich konfiguračních souborech. Nástroj slouží výhradně ke tvorbě hlášení informací pro cestující, která jsou ve všech typech vozů stejná.

## 4.5. Formát souboru hlásiče

Soubor hlásiče má příponu .etanc („Eurotram announcements“) je sestaven hlavičky, seznamu základních hlášení seznamu zastávek. Zastávky nejsou nijak tříděny.

Hlavička je uložena následovně:

ETAN D J...J0 A...A0 P...P0 L...L (N...N0)...(Ni...Ni0) Z

První byty odpovídají textu „ETAN“ kódování ASCII. je 16-bitové číslo označující verzi formátu souboru. je unixový čas reprezentovaný jako 64-bitové číslo. Řetězec označuje název hlásiče. Řetězec označuje autora hlásiče. Řetězec označuje složku ve které jsou obsaženy zvukové soubory označující linky definované  $L N$ . Všechny tři řetězce jsou uloženy kódování UTF-8. označuje počet linek  $L$ , pro které jsou simulátoru obsaženy hlášení. i jsou 8-bitová čísla. 8-bitové číslo označuje počet řetězců označujících názvy nečíselných linek. je 32-bitové číslo označující počet zastávek.

Po hlavičce je souboru obsažena informace základních hlášení. Každé takovéto hlášení je uloženo jako řetězec znázorněný jako sloužící jako odkaz na soubor hlášení. Pořadí hlášení je jasně dané je následující: znělka, hlášení příští zastávky, hlášení informující že příští zastávka je na znamení, hlášení informující že tato zastávka je na znamení, hlášení výstupu do vozovky, hlášení konečné zastávky českém anglickém jazyce, hlášení přestupu na metro, hlášení přestupu na vlaky (linky S), hlášení změně trasy linky, hlášení upozornění pro cestující, hlášení „Stanice metra...“ „...je uzavřena.“, hlášení jízdy do vozovny („Vlak jede do vozovny“), hlášení „ze zastávky“ „je dále vedena jako linka“, slovo „směr“, hlášení uzávěru lanové dráhy na Petřín služební hlášení odjezdu konečné za jednu minutu.

Každá zastávka je uložena následovně:

SSSS N...N0 A...A0 L...L (N...N0)...(Ni...Ni0) F

S je 4-bytový staniční kód kódování ASCII, řetězec je název stanice. Řetězec je název souboru zvukového hlášení. Složka je udána jako relativní souboru simulátoru. Dále následuje 8-bitových čísel znázorňující linky projíždějící zastávkou  $L$ . je číslo 8-bitové. 8-bitové číslo označuje počet řetězců označujících názvy nečíselných linek, které zastávkou projíždí. je 8-bitové bitové pole označující flagy zastávky – zda je přítomen přestup na metro, přestup na vlaky (hlášení „přestup na linky S“) zda je potřebné hlásit uzávěr lanové dráhy na Petřín. Všechny řetězce jsou zakódované ve formátu UTF-8.

Všechna čísla jsou se znaménkem podporují záporné hodnoty za pomocí dvojkového doplňku. Mezery jsou čistě grafickým dodatkem nejsou ve formátu (mimo řetězce) nikdy obsaženy. Čísla jsou uloženy ve formátu little-endian. Cesty souborům jsou relativní jsou

Ročníková práce

**Fyzicky realistický simulátor pražských tramvají včetně okolí bodovacího systému**

procházeny od umístění aplikace simulátoru. Koncovky souborů nejsou ukládány – všechny zvukové soubory jsou ve formátu WAV.

## 5. Implementace simulací

### 5.1. Infrastruktura působení sil na vozidla

Fyzikální simulace jsou aplikovány pro každé současně viditelné vozidlo. Jelikož mají normální, „grafické“, snímky mezi sebou nepravidelné intervaly, je přidán systém tzv. „fyzikálního snímku“, který volá funkci zpracovávající fyzikální simulace každých 20 milisekund neboli 50-krát za sekundu. Tato hodnota je simulátoru nastavitelná. Pro každé vozidlo se každý fyzikální snímek spustí funkce vypočítávající všechny simulované veličiny. Vozidla ukládají svojí hmotnost, současnou rychlost, plochu jednotlivých stran, maximální výkon motorů, celkovou spotřebovanou energii ale jiné fyzikální hodnoty potřebné pro výpočty.

Na vozidlo jsou aplikovány výsledné síly – pohybová (působící ve směru jízdy, síla pohybuje brzdí tramvaj) vertikální (použitou pro simulaci vykolejení). Ostatní síly jsou uloženy pro účely výpočtů těchto výsledných.

Pohybová síla skládá za síly motorů společné odporové síly. simulátoru je také vypočítávána maximální odporová síla  $F_{rmax}$ , která je rovna podílu součinu rychlosti a hmotnosti  $m$ , intervalu volání fyzikální funkce  $t=1/50$ .

$$\begin{aligned}v &= at \\v &= \frac{F_{rmax}}{m} t \\F_{rmax} &= vm/t\end{aligned}$$

Síla  $F_{rmax}$  určuje potřebnou sílu pro zabrzdění tramvaje za jedno volání funkce. To znamená že odporové síly nemohou překračovat tuto sílu, jelikož by následně zrychlovali tramvaj do opačného směru.

Síly fyzikální vlastnosti pro několika-článekové tramvaje či několika-vozové soupravy jsou vypočítávány pouze jednou.

### 5.2. Tíha

Tíhová síla  $F_g$  je součin součtu hmotnosti tramvaje  $m_t$  tíhového zrychlení  $g$ .

$$\vec{F}_g = m_t \times \vec{g}$$

Tíhové zrychlení je bráno jako  $9,81 \text{ ms}^{-1}$ . Má-li souprava více vozů, je hmotnost násobena. Hmotnost tramvaje je možné nastavit konfiguračním souboru typu vozidla.

### 5.3. Tření, valivý odpor prokluz kol

Síla valivého odporu je rovna součinu síly normálové  $F_n$  koeficientu valivého odporu  $c_{rr}$ .

$$\vec{F}_r = \vec{F}_n c_{rr}$$

Koeficient valivého odporu je simulátoru vždy nastaven na hodnotu 0,001, což je rovno odporu železných kol na železné koleji.<sup>[29]</sup>

Třecí síla  $F_t$  je rovna součinu síly normálové  $F_n$  (která je rovna  $F_g \cos \theta$ , kde  $\theta$  je rovna úhlu nakloněné roviny které se současně vůz nachází) součinitele smykového tření  $c_f$ .

$$\vec{F}_t = \vec{F}_n c_f$$

Změny počasí jiné podmínky mění pouze koeficient tření. Koeficient tření za ideálních podmínek je nastaven na 0,5. Koeficient tření za ideálních podmínek během klidu vozidla je nastaven na 0,7.<sup>[30\*]</sup> Různé počasí tento koeficient avšak mění.

Stav	Změna koeficientu tření
Děšť nebo Sníh	-0,2
Náledí (zimní roční období)	-0,3
Listí (podzimní roční období)	-0,2

Tabulka 8: Efekt počasí různých stavů na koeficient tření

Třecí síla je aplikována pouze při nouzovém brzdění vozidla kolejnicovou brzdou. Při nastavení řadiče jízdy do záporné pozice používá vozidlo brzdění regenerační, které je popsáno detailněji kapitole 5.4.

Prokluz nastává kdy výsledná pohybová síla je ostře větší než síla třecí.<sup>[31]</sup> takovém případě simulátor neaplikuje dále pohybovou sílu dokud není prokluz vyřešen.

## 5.4. Síla působení motorů rychlost vozidla

Při jízdě vozidla je vypočítávána jeho síla působení motorů. Výslednice této síly, síly třecí síly tíhové tvoří celkem sílu pohybovou.

Síla působení motoru  $F_f$ , jinak označována jako trakční usilí, je rovna podílu součinu výkonu motorů a procenta použitého výkonu  $e$ , současné okamžité rychlosti  $v$ .

$$\vec{F}_f = Pe/v$$

Je-li okamžitá rychlost rovna nule, není možné důvodů dělení nulou tento vztah použít. Pro sílu motoru při rozjezdu, který je zde definován jako  $v < 3 \text{ m/s}$ , jsou použity následující vztahy:

$$M = P/\omega$$

$$\vec{F}_f = \frac{M \eta p^{[32]}}{r}$$

$M$  je točivý moment motoru, který je roven podílu výkonu motoru a jeho úhlové rychlosti  $\omega$ . Výsledná síla působení motoru je potom vypočítána vynásobením točivého momentu motoru, jeho účinnosti (v simulátoru vždy rovna 95%), hodnoty poměru převodů a následným vydělením tohoto součinu poloměrem kol.

Výsledná rychlost pohybu vozidla je určena na základě 2. Newtonova pohybového zákona, dle kterého sílu je možné zapsat jako součin hmotnosti zrychlení, což lze vyjádřit také rovnicí  $F = ma$ . Zrychlení se následně převede na okamžitou rychlost  $v$ , která je následně poslána do grafické části simulátoru.



## 5.5. Síly oblouku vykolejení

Během zatáčení vozu oblouku je důležitý poměr mezi silou vertikální a silou boční  $L$ .

$$n_f = \frac{\vec{L}}{\vec{V}}$$

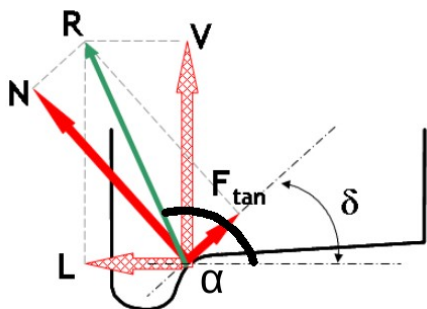
Je potřeba také vypočítat tzv. „hraniční poměr“ po kterém se kolo pohybuje buď nahoru po kolejnici nebo po ní sjíždí. Rovnice určující stav při nejvyšším při kterém se kola vertikálně nepohybují se jmenuje Nadalova rovnice je definována následovně:

$$\frac{\vec{L}}{\vec{V}} = \frac{\tan(\delta) - c_f}{1 + c_f \times \tan(\delta)} \quad [33]$$

$L$  je vertikální síla, která je rovna vertikální složce síly tíhové,  $V$  je síla boční, simulátoru dostředivá.

Výsledná síla nadnesení kola nad kolejnicí  $F_{tan}$  je rovna součinu výsledné síly a arcus kosinu rozdílu úhlu doteku a úhlu vektoru síly k vodorovné ose  $\alpha$ , která je definována jako součet sil a  $V$ . Vertikální složka této síly je vypočítána jako součin síly  $F_{tan}$  kosinu úhlu  $90 - \delta$ .

$$\begin{aligned} \vec{R} &= \vec{L} + \vec{V} \\ \vec{F}_{tan} &= \vec{R} \times \cos(|\alpha - \delta|) \quad [33] \\ \vec{F}_{tan} &= \vec{F}_{tan} \cos(90 - \delta) \end{aligned}$$



Obrázek 15: Hodnoty použité při výpočtu síly nadzvednutí vozidla oblouku<sup>[32]</sup>

Pokud výška kol nad tratí překročí výšku trati, považuje se vozidlo za vykolejené. Při vykolejení se program ihned vypne vypíše hlášku vykolejení vozidla.

## 5.6. Aerodynamická odporová síla

Aerodynamická odporová síla je vypočítána následujícího vztahu:

$$\vec{F}_{\text{odporová}} = \frac{1}{2} \rho v^2 c S^{[34]}$$

$v$  je okamžitá rychlost vozidla, obou rovnicích určuje  $\rho$  hustotu vzduchu (která je rovna  $1\text{kg/m}^3$ ), je činitel odporu (definován konfiguračním souboru vozidla)  $S$  plochu průřezu čela vozidla metrech čtverečných. Hodnota je vypočítána výšky šířky uvedené konfiguračním souboru. Síla bočního větru je zanedbána.

## 5.7. Elektrické systémy vozidel celková práce vozidla

U elektrických systémů je simulován kontakt pantografu trolejovým vedením, současné napětí úseku sítě trolejového vedení, spotřeba elektrické energie tramvají celková spotřeba elektrické energie za celou jízdu.

Pro výpočet spotřeby energie je použit následující vztah:

$$E = P/t$$

$P$  je rovno výkonu motoru danou chvíli, je vypočítáno jako maximální výkon motorů procentuální zátěž nastavenou řidičem  $c \in \langle 0; 1 \rangle$ . Čas je konstantní je roven časovému intervalu mezi jednotlivými voláními funkcí fyzického výpočtu. Pro každý fyzikální interval se energie přičítá. Při brzdění, které je simulátoru zjištěno dle záporného příkonu, se energie vrací zpátky, to účinností 75% ( $\eta = 0,75$ ).

Pro výpočet současného proudu je využit Wattův zákon.

$$P = VI \cos \phi$$

$V$  je současné napětí, je současný proud  $I$  je fázový úhel střídavého proudu, který je simulátoru nastaven na 0,9 radiánů. Současné napětí je nastaveno na napětí sítě, neboli 600V.<sup>[5]</sup>

Celková energie využita vozidlem je později použita pro výpočet celkových emisí, viz kapitola 5.10 „Výpočet srovnávání emisí“.

## 5.8. Osvětlení

Pro simulaci osvětlení je použita modifikovaná verze Blinn-Phongova modelu odražení světla. Osvětlení Blinn-Phongova modelu je možné rozdělit na části – světlo okolní (angl. ambient), světlo difuzní (angl. diffuse) světlo odražené (angl. Specular). Blinn-Phongův model je pouze malá úprava původního Phongova modelu. Jediná změna se týká výpočtu odraženého světla tím je přidání polovičního vektoru pro lepší zobrazení při velkém rozdílu uhlů vektoru vzhledu vektoru světla. <sup>[20 p. 115-125 \*]</sup>

V standardní verzi modelu jsou přítomny pro každou část vlastní světelné hodnoty. důvodu způsobu exportování souboru modelovacím programem, je pro každé těleso nastavena pouze barva základní.

Výsledná barva fragmentu je rovna součinu základní barvy  $c_{r,g,b}$  větší hodnoty (funkce  $\max(a, b)$  – vrací „maximum“ dvou hodnot) mezi hodnotami světla okolního a součtu sum výsledků všech světél – sum jejich difúzních ( $d_{r,g,b}$ ) odrazových částí  $s_{r,g,b}$ .

$$[o_r, o_g, o_b] = [c_r, c_g, c_b] \times \max\left(a, \sum_{i=1}^n [d_r, d_g, d_b] + \sum_{i=1}^n [s_r, s_g, s_b]\right)$$

$$\text{kde } a \in \langle 0, 1 \rangle, c_{r,g,b} \in \langle 0, 1 \rangle, d_{r,g,b} \in \langle 0, 1 \rangle, s_{r,g,b} \in \langle 0, 1 \rangle$$

Barva difuzního světla je součinem silou difúzního světla  $p_d$ , barvou světla  $l_{r,g,b}$  atenuací  $t_d$ .

$$[d_r, d_g, d_b] = p_d \times [l_r, l_g, l_b] \times t_d \text{ kde } p_d \in \langle 0, 1 \rangle, l_{r,g,b} \in \langle 0, 1 \rangle, t_d \in \langle 0, 1 \rangle$$

Barva světla odraženého je součinem čtyř částí – silou odraženého světla  $p_s$ , barvy odraženého světla  $l_{r,g,b}$ , odrazivost materiálu  $r_m$  atenuace světla  $t_s$ .

$$[s_r, s_g, s_b] = r_m \times p_s \times [l_r, l_g, l_b] \times t_s \text{ kde } r_m \in \langle 0, 1 \rangle, l_{r,g,b} \in \langle 0, 1 \rangle, t_s \in \langle 0, 1 \rangle$$

Síla difúzního světla  $p_d$  je skalárním součinem normalizovaných normálů  $n_f$  obličeje na kterým fragment leží normalizovaného vektoru směřujícího na světlo. Součin je zdola omezen na hodnotu funkcí  $\max$ . Výpočet je graficky znázorněn na následujícím obrázku.

$$p_d = \max(\hat{n}_f \cdot \hat{u}, 0)$$

Síla odraženého světla  $p_s$  je skalárním součinem normalizovaných normálů  $n_f$  obličeje na kterým fragment leží středového vektoru  $h$ . Vektor je roven normalizovanému součtu vektoru směřujícího na světlo a vektoru směřujícího od pohledu uživatele  $v$ . Výsledek je taktéž umocněn na exponent  $s_m$  uvažující hodnotu lesku materiálu. Výpočet je graficky znázorněn na obrázku 8.

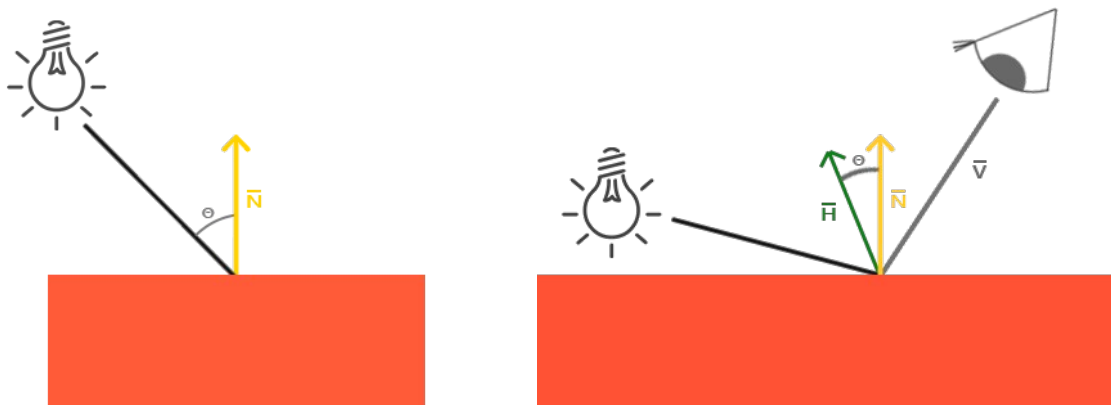
$$\vec{h} = \vec{u} + \vec{v}$$

$$p_s = \max(\vec{h} \cdot \vec{n}_f, 0)^{s_m}$$

Odrazivost materiálu  $r_m$  je vypočítávána tzv. indexu lomu („Index of Refraction“, IOR) označeného jako  $i_r$  to následující rovnicí:

$$r_m = ((i_r - 1) / (i_r + 1))^2 * 12,5 \text{ kde } i_r \in \langle 1, \infty \rangle^{[35]}$$

Odrazivost je následně omezena na interval  $r_m \in \langle 0, 1 \rangle$  pro splnění podmínky rovnice pro výpočet barvy světla odraženého.



Obrázek 16: Znázornění výpočtu difúzního světla dle (Blinn-)Phongova modelu.

Obrázek 17: Znázornění výpočtu odraženého světla dle Blinn-Phongova modelu.

Obrázky jsou uspořádány dle čísla. Obrázek menším číslem je vlevo, větším vpravo.

Zdroj prvního obrázku: 20 p. 116, druhého: 20 p. 122

Atenuace světla je vypočtena následující rovnicí:

$$t_x = \frac{1}{c + ld + qd^2} \text{ kde } t_x = \langle 0, 1 \rangle, c = \langle 1, \infty \rangle, l, q = \langle 0, \infty \rangle, d = \langle 0, 1 \rangle^{[20 \text{ p. } 140]}$$

Dosah světla (v bezrozměrných jednotkách)	Konstantní parametr $c$	Lineární parametr $l$	Kvadratický parametr $q$
7	1	0,7	1,8
20	1	0,22	0,2
50	1	0,09	0,032
100	1	0,045	0,0075
200	1	0,022	0,0019

Tabulka 9: Hodnoty pro výpočet atenuace závislosti na dosahu světla.<sup>[20 p. 141]</sup>

V rovnici jsou  $c$ , a konstanty určené pro danou vzdálenost světla Tabulky 2. označuje vzdálenost.  $t_x$  označuje výslednou atenuaci.

V simulátoru jsou implementovány typy světel – světlo směrové (angl. Directional), světlo bodové (angl. Point) světlo reflektorové (angl. Spotlight). Síla odraženého světla se všech typů světel vypočítává stejně.

Světlo směrové je tří typů nejjednodušší. Směr světla je předem nastaven programem je vždy stejný. Pro směrové světlo také není vypočítávaná atenuace, jelikož směrové světlo nemá definovanou žádnou polohu.<sup>[20 p. 140 \*]</sup>

Směrový vektor pro světla bodová reflektorová je roven normalizovanému rozdílu polohy světla lokace fragmentu. <sup>[20 p. 137 \*]</sup>

Světlo bodové se od směrového liší využitím vypočítaného směrového vektoru vypočítáváním atenuace světla.

Světlo reflektorové je ze tří typů pro výpočet nejtěžší. Obsahuje veškeré výpočty světla bodového k tomu vypočítává sílu světla závislosti od vnitřního vnějšího úhlu kuželu. Sílu reflektorového světla je možné vyjádřit následující rovnicí, kde úhel určuje úhel mezi vektorem směru světla a vektorem ukazujícím na polohu světla pohledu fragmentu  $s$ . Hodnoty  $a$  určují vnitřní vnější úhly kuželu reflektorového světla. Vektor směru světla je otočen (vynásoben -1) aby byl nasměrován do směru světla, nikoliv naopak. <sup>[20 p. 143 \*]</sup>

$$a = s \cdot \hat{d}$$

$$p = \frac{a - o}{i - o} \text{ kde } o > i, i, o, a \in (0, \frac{1}{2}\pi) \text{ rad} \Leftrightarrow (0, 90) \text{ deg}$$

Rovnice vypočítává podíl rozdílu uhlu a vnějšího uhlu a rozdílu vnitřního vnějšího uhlu. Hodnota čitateli je vydělena rozdílem vnitřního uhlu vnějšího uhlu. Hodnota jmenovateli je vždy kladná, jelikož  $\cos(i) > \cos(o)$ . Hodnota je po výpočtu omezena na interval  $p \in (0, 1)$ , což znamená že jakákoliv záporná či nulová hodnota jmenovateli.

Pokud je  $\geq$  (úhel je větší nebo roven než maximální, vnější úhel světla), s tím je kosinová hodnota  $\cos(a) < \cos(o)$ , vychází čítec záporně či nulově, jelikož je hodnota ve jmenovateli vždy kladná, daný fragment není světlem osvětlen. (Výsledek je záporný což se následně omezí na hodnotu 0.)

Funkce skalárního součin vrací úhel podobě kosinové hodnoty. pohledu časové účinnosti je toho důvodu jednodušší počítat kosinovými hodnotami uhlů než jejich opravdovými hodnotami. Veškeré výpočty probíhají radiánech.

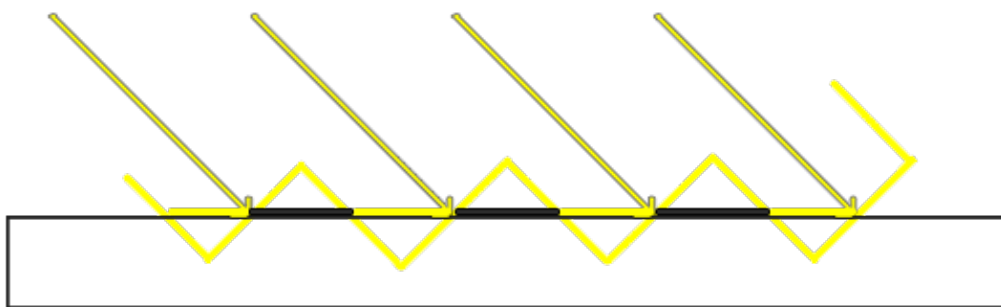
Pro korekci gammy (poměru mezi fyzikálním jasem jasem vnímaným člověkem) byl využit hardwarový komponent přes API OpenGL. <sup>[20 p. 277 \*]</sup>

## 5.9. Stíny

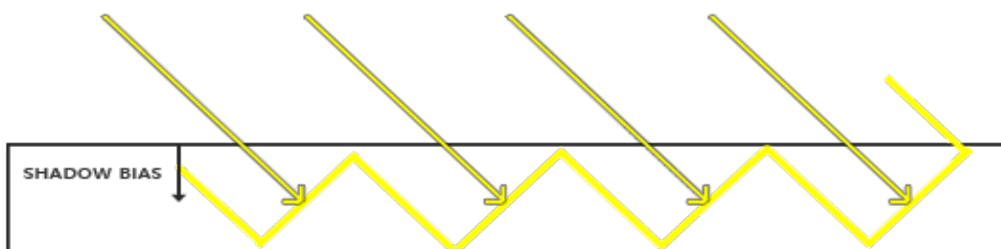
Pro simulaci stínů je použita interní „hloubková mapa“ API OpenGL. Hloubková mapa určuje které objekty jsou daného pohledu (například uživatele, světla apod) zakryté nejsou tím pádem vidět. Hloubková mapa je pro tento účel správným nástrojem jelikož stíny se vždy nachází tam, kam nedosahuje světlo, neboli zakrytá místa pohledu daného světla. Hloubkové mapy pro každé světlo se vykreslují na hloubkovou texturu, která obsahuje hloubkovou informaci fragmentů jenž dané světlo vidí. Tato hloubková textura ukládá také vzdálenost nejbližšího fragmentu ke světlu. Všechny texturey světel jsou následně posílány do shaderu, kde jsou následně přečteny aplikovány.

Hloubka současného fragmentu (která je zjištěna aplikováním transformační matice světla na pozici fragmentu uloženou jakou čtyřmístný vektor dělením výsledné pozice posledním parametrem vektoru) je porovnána hloubkou nejbližšího světla fragmentu. Je-li současná hloubka vyšší než hloubka nejbližšího fragmentu, nachází se daný fragment ve stínu. Výsledný stín je průměr stínu fragmentu jeho okolních fragmentů. Způsoby přidávání plynulejších přechodů stínů se nazývají PCF („percentage-closer filtering“), tento termín avšak zahrnuje velký počet různých způsobů tvorby plynulejších přechodů, průměrování okolních fragmentů je pouze jeden nich. [20 p. 283-299 \*]

Pro prevenci tzv. „stínového akné“ je aplikována hodnota (tzv. „shadow bias“), posouvající hodnotu hloubky několik milimetrů dovnitř do stěny. Tímto jsou eliminovány nepřesnosti modelu, vznikající kvůli ohraničenému rozměru hloubkové textury kvůli čehož několik fragmentů může dostat stejnou přiřazenou hloubku. [20 p. 307 \*]



Obrázek 18: Dopad světla pod úhlem bez aplikování posuvné hodnoty



Obrázek 19: Dopad světla pod úhlem po aplikování posuvné hodnoty (tzv. „bias“)

Pro každé světlo, které má simulovaný stín, je potřeba znovu vykreslit hloubkovou informaci scény pohledu pozice světla. Toto vykreslování je sice rychlejší než vykreslování normální scény, jelikož se nepočítají výsledné barvy, avšak je to stále náročná operace. tohoto důvodu jsou vykreslovány stíny pouze pro nejpodstatnější světla, to stíny ze světla nebe (světlo pocházející slunce měsíce), stíny ovladatelného osvětlení (baterky řidiče) předních světel tramvaje.

Stíny jsou simulovány pouze pro směrová reflektorová světla. Pro světla bodová, která svítí do všech stran rovnoměrně, je potřeba vykreslovat scénu krát to do každého směru.

## 5.10. Simulace zvuků

### 5.10.1. Simulace zvuku motoru brzd

Simulace zvuků motoru brzdění tramvají je založena na několika nahraných zvucích rozjezdu brzdění. závislosti na nastavený výkon motoru se dokola přehrává sekce zvuku, obvykle 200 milisekund dlouhá. Aby nebylo slyšet přechod mezi jednotlivými opakováními zvuku, zvuk se pokaždé náhodně modifikuje to následujícími způsoby:

- Použije se otočený zvuk (ten, který se hraje od konce). Tyto zvuky jsou předem vytvořeny nějakém zvukovém editoru, jelikož tvoření takových zvuků během práce programu je příliš početně náročné.
- Prolnou se sousedící zvukové „bloky“.

TODO přidat tabulku diagram otáčení manipulování zvuku vč. Spektrogramu bez manipulace po manipulaci + vyznačit klikání

*Obrázek 20: Spektrogram neupraveného opakujícího se zvuku*

*Obrázek 21: Spektrogram upraveného opakujícího se zvuku*

Z spektrogramů zvukových vzorků je možné jednoznačně usoudit, že úprava zvuku simulátoru odstraňuje zvukový efekt „klikání“ ve zvuku. TODO dopopsat



### **5.10.2. Simulace zvuku jízdy**

Simulace zvuku jízdy je sestavena několika prvků – zvuku jízdy po kolejnicích na rovných úsecích, zvuky tření při odbočování ze zvuků přejezdu výhybky.

### **5.10.3. Simulace jiných zvuků**

Simulace ostatních zvuků typu varovného signálu zavírání dveří, zvuků palubního počítače, zvuku znamení řidiči či zvuku zvonění jsou nahrány předem závisí na jednotlivém typu vozu. Jednotlivé souborové cesty zvuků jsou specifikovány konfiguračním souboru typu vozidla, který je přesněji popsán kapitole 1.3 „Konfigurace vozidel“.

### **5.10.4 Simulace hlášení**

V simulátoru je obsažena simulace hlášení, to jako hlášení pro cestující tak služební. Hlášení, či některých případech jejich části, jsou nahrány předem uloženy jako soubory ve formátu WAV. simulátoru se následovně hrají předem určeném pořadí vytvořeném dle reálných hlášení. Toto odpovídá reálnému systému hlášení autobusech tramvají<sup>[1][2]</sup>. Hlášení je možné simulátoru jakkoliv kombinovat či opakovat. Detaily ohledně hlášení na jednotlivých zastávkách jsou uloženy ve speciálním souboru hlásiče jehož formát je popsán kapitole 4.5 „Formát souboru hlásiče“.

## 5.11. Výpočet srovnávání emisí

### 5.11.1. Výpočet emisí automobilové dopravy

Výpočet celkových emisí, které by vyprodukovala individuální automobilová doprava, je roven součinu potřebných automobilů (podílu maximálního počtu cestujících kapacity automobilu) délky linky, která je vypočítána sumě všech délek tratí.

$$e_{auto} = \frac{P_{max}}{C_{auto}} s$$

Emise průměrného automobilu na km činí 400gCO<sub>2</sub>e.<sup>[36]</sup> Emise automobilové dopravy jsou vypočítány pro různé kapacity – kapacity průměrné (hodnota 1,25 vybrána arbitrárně dle vlastního pozorování – většina automobilů má 1, občas cestující) kapacity maximální, což pro průměrný automobil činí osob.

### 5.11.2. Výpočet emisí tramvaje

Výpočet emisí tramvaje je vypočten na základě průměrných emisí při produkci elektrické energie celkové spotřeby elektrické energie tramvají. Celková elektrická spotřeba tramvaje je vypočítávána při fyzikální simulaci. Emise při výrobě kilowatt-hodiny (3 600 000 joulů/J) elektrické energie ČR činí 450gCO<sub>2</sub>e. Jsou použita data pro rok 2023.<sup>[37\*]</sup>

$$e_{tram} = P / 3600 J \times 450 gCO_2 e / kWh$$

### 5.11.3. Výpočet ušetřených emisí

Ušetřené emise se rovnají rozdílu hodnot emisí automobilové dopravy emisí tramvaje. Je přidán také výpočet ceny energie pro Českou republiku, Polskou republiku Spolkovou republiku Německo. Cena za kilowatt je aktuální 1. lednu 2024. Ceny jsou vypočítány eurech, převedeny jsou dle denního kurzu České národní banky (ČNB) dle kurzu aktuálnímu ke 21. březnu 2025.<sup>[39]</sup>

## 6. Simulace počasí jeho účinků

### 6.1. Infrastruktura systému počasí

Infrastrukturu systému počasí je možné podělit na části – část grafická část fyzikální. Část fyzikální je složena ze změn fyzikálních koeficientů následně použitých při výpočtu sil rychlostí. Část grafická vykresluje realistický grafický efekt odpovídající stavu počasí. Jakoukoliv dobu může být přítomno několik stavů počasí.

### 6.2. Denní noční cyklus

Denní noční cyklus je simulátoru nastaven na pevnou dobu závislost na roční období.

### 6.3. Počasí

#### 6.3.1. Vítr

Větrná podmínka mění pouze aerodynamickou odporovou sílu. Vítr nemá současně žádný vliv na grafickou podobu simulátoru.

#### 6.3.2. Déšť

Déšť mění koeficient tření kontaktní odpor mezi sběračem trakčním vedení. Změny koeficientů závisí na síle jevu – simulátoru je toto dáno jako počet kapek.

Déšť také obsahuje grafický efekt kapek, ztemnění okolí (aproximace změny jasu při zanesení oblaky) zešeření oblohy.

#### 6.3.3. Bouřky

Bouřka se simuluje rychlou změnou barvy oblohy za pomoci přednastavených hodnot. Hodnotu není možné mimo simulátor měnit.

#### 6.3.4. Mlha

Efekt mlhy je vytvořen snížením viditelného pole za pomoci nastavení parametrů kamery (parametru tzv. „far plane“ neboli vzdálenost po které se již objekty nevykreslují) zešeřením oblohy. Zešeření oblohy je vytvořeno násobením stavu dne (konstanty okolního světla, viz kapitola 5.7 „Osvětlení“) výchozí šedé barvy. Existuje také náhodná šance (současně je simulátoru nastavená na 0,1%) že místo mlhy bude aplikována písečná bouře. Při tomto stavu je výchozí barva mlhy bude nastavena na barvu hnědou.

## 6.4. Počasí závisující na ročním období

### 6.4.1. Sníh

Sníh je simulátoru modifikovanou verzí deště. Využívá stejné grafické infrastruktury jako dešť, mění stejné koeficienty. Mění také barvu oblohy – nastavuje jí na světlejší.

## 6.5. Roční období

### 6.5.1. Jaro léto

Během jara léta je simulátor standardním režimu jeho standardními texturami. Toto je výchozí stav simulátoru. Jedinými změnami mezi jarem létem jsou časy denního nočního cyklu.

### 6.5.2. Podzim

Během podzimu se mění všechny textury stromů trávy na odpovídající textury. Soubory textur použité během podzimního ročního období jsou označeny příponou *Autumn* např. *tp-grassAutumn.png*.

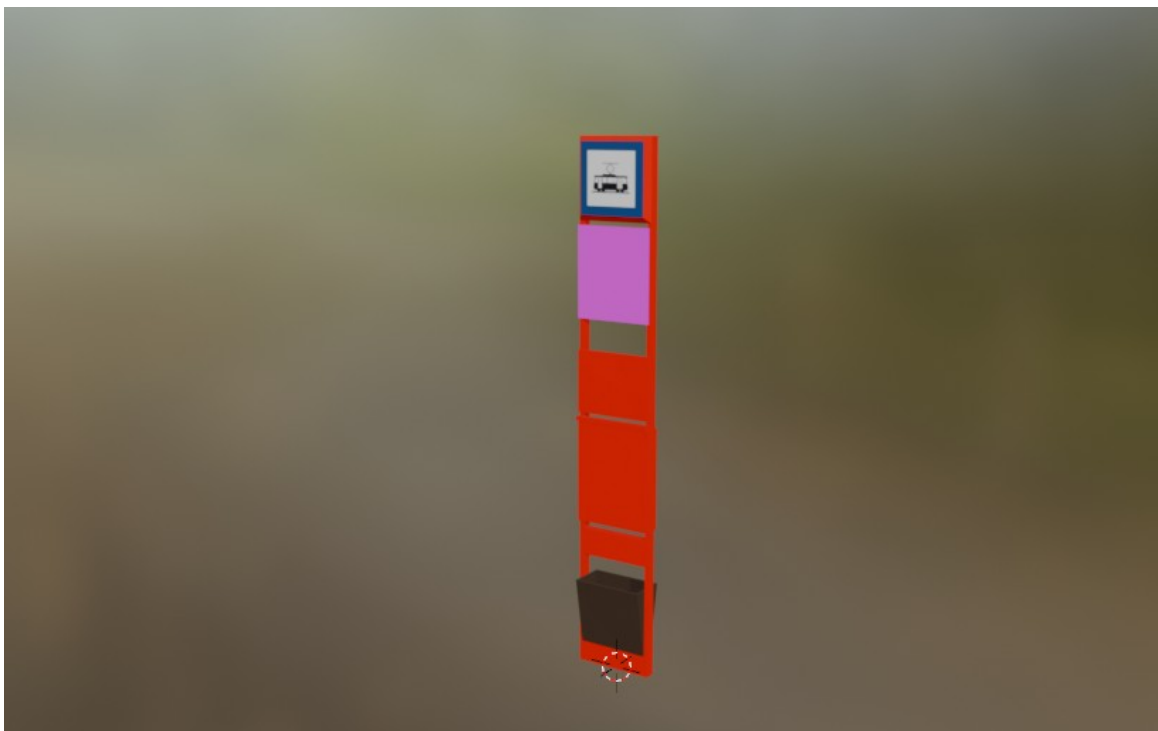
### 6.5.3. Zima

V zimě je aplikováno nejvíce grafických efektů. Střechy všech nepohybujících se objektů jsou vykreslovány bílou barvou. Soubory textur pro zimní roční období jsou označeny příponou *Winter* např. *tp-grassWinter.png*.

## 7. Modelování okolí

### 7.1. Zastávky

Zastávky jsou simulátoru graficky označeny staničním sloupem. Staniční sloup je modelován dle standardních vnitropražských označníků PID používaných na tramvajových zastávkách.



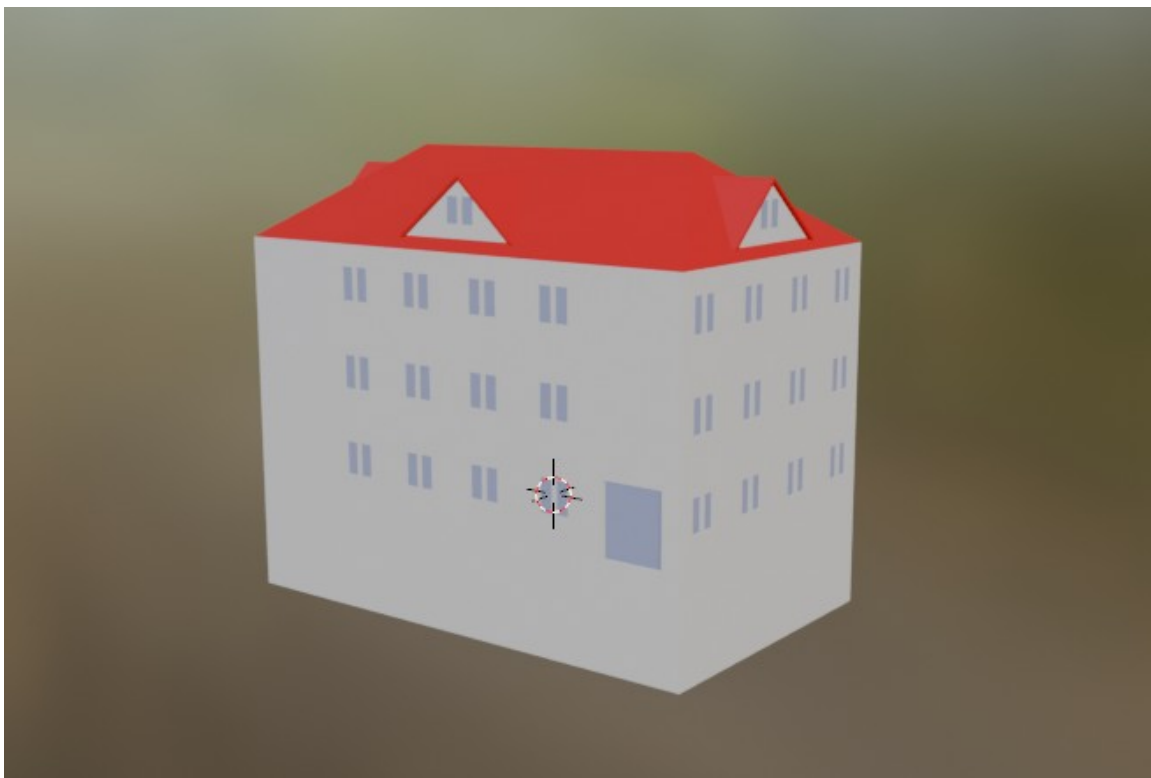
Obrázek 22: Obrázek modelu zastávkového sloupu

### 7.2. Standardní budovy

#### 7.2.1. Typy budov

Simulátor obsahuje typy obvyklých budov. Každá budova má jiné, avšak předem definované rozměry.

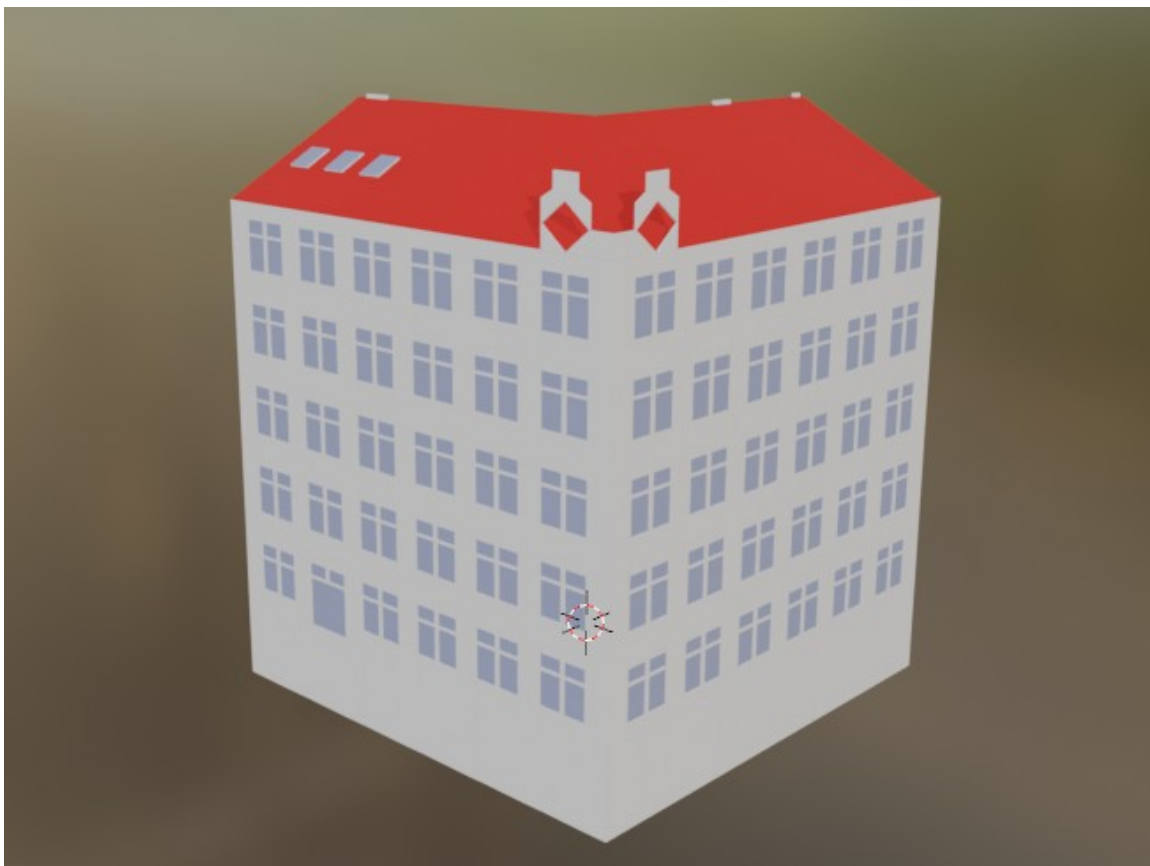
Každá budova, která má okna, musí obsahovat materiál názvem „Material.windowDay“. Tento materiál je při rozsvícení nočního osvětlení přestaven na „Material.windowNight“. Tato změna povoluje simulovat noční osvětlení uvnitř budov.



*Obrázek 23: Obrázek modelu rodinného domku*



*Obrázek 24: Obrázek modelu vícepatrového řadového domu*



*Obrázek 25: Obrázek modelu vícepatrového rohového řadového domu*



*Obrázek 26: Obrázek modelu panelového domu*

### 7.2.2. Systém obarvení budov

Pro přidání realismu je při setmění změněn materiál oken. Při této změně je nastavena barva vnitřního osvětlení budovy. Pro různé části budov jsou použity různé barvy. Zde jsou uvedeny jejich hexadecimální kódy, slovní názvy, názvy použití a názvy materiálů budovy.

Hexadecimální kód barvy	Slovní název barvy	Názvy použití	Název materiálu
A5AED5	Světlá modrá, až tyrkysová	Okna, denní	Material.windowDay
E7E213	Světlá, skoro plně žlutá	Okna, noční	Material.windowNight
E7000D	Světle červená, barva čisté střechy	Střecha	Material.roof

Tabulka 10: Barvy materiály budov



## 8. Systém bodování

### 8.1. Seznam akcí za které se body odečítají

Akce	Počet odečtených bodů
Vykolejení	1000
Projetí zastávky na znamení cestujícími	500
Převýšení rychlostního limitu	25 za vteřinu
Odjetí kontrolního bodu před dobou odjezdu	250
Neodhlášení zastávky včas (tj. Před odjezdem ze zastávky)	100

Tabulka 11: Seznam akcí za které se body odečítají

### 8.2. Seznam akcí za které se body přičítají

Akce	Počet přičtených bodů
Zastavení zastávce	500
Zastavení zastávce na znamení	1000
Projetí výhybky	200
Odhlášení zastávky	200

Tabulka 12: Seznam akcí za které se body přičítají

### 8.4. Výpočet výsledné známky

Výsledná známka je určena podílem dosažených bodů maximálního dosažitelného počtu bodů. Podíl se následně převede na známku dle programu uložené tabulky. Výchozí nastavení je standardní české známkování od jedné do pěti, kde známka je nejlepší 5 nejhorší. Tabulka, která definuje procentový interval názvy známek je možné upravovat.

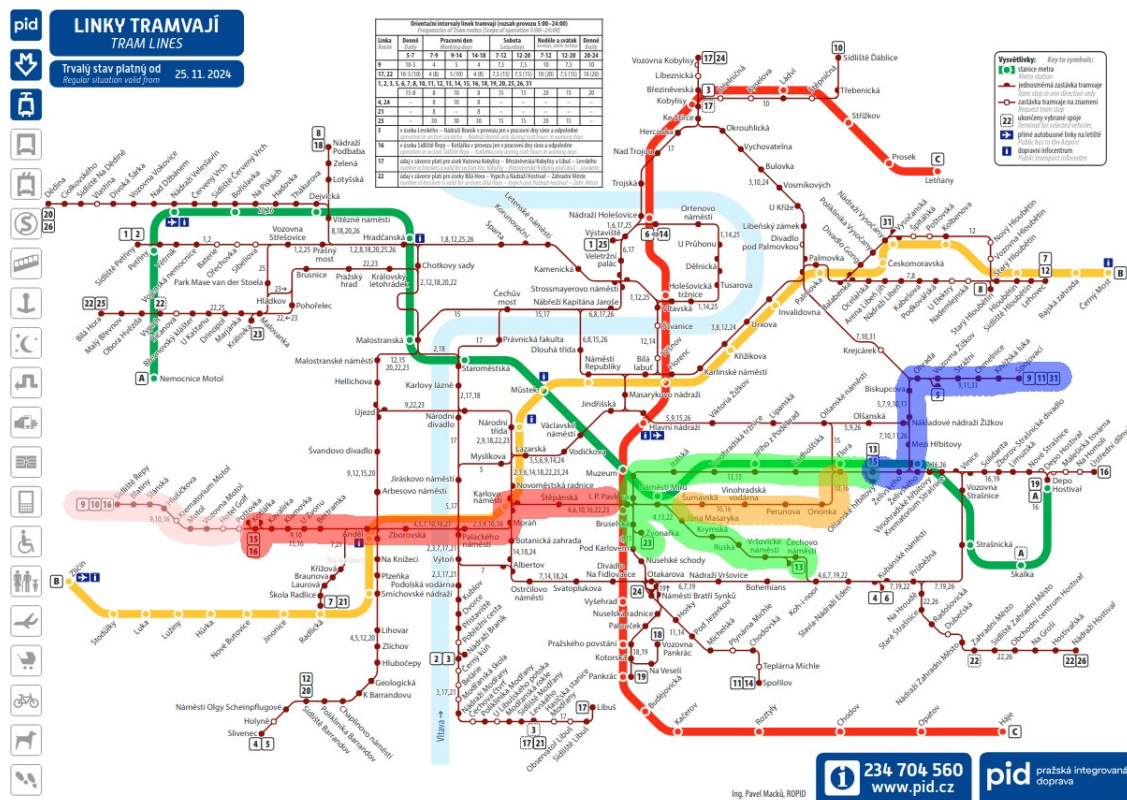
## 9. Další rozvoj projektu

### 9.1. Modelování dalších vozidel

Jakékoliv budoucí rozvinutí projektu by potřebovalo přidat jiné typy tramvajových vozů než pouze typ T3R.P. Nejjednodušší vůz na přidání by byl vůz T3R.PLF, jelikož se jedná nízkopodlažní variantu nevelkým počtem změn od původního vozu. Dále by se měl vymodelovat vložit vůz typu KT8D5.RN2P důvodů jeho schopnosti obousměrné jízdy. Po přidání tohoto typu by bylo možné vymodelovat další úseky tramvajových tratí, například úsek Levského – Libuš, či simulovat jakoukoliv výluku kdekoli na síti.

### 9.2. Modelování dalších tramvajových tratí

V budoucnu by měli být přidány další tramvajové tratě navazující na tratě současně simulátoru obsažené. první řadě by se tak jednalo úsek Olšanské hřbitovy – Želivského – Biskupcova – Vozovna Žižkov – Spojovací či úsek Náměstí Míru – Flora přes zastávku Šumavská. Přidáním těchto tratí bude možno simulovat většinu linky 11 (od konečné linky na zastávce Spojovací až do zastávky Bruselská) značnou část linky 10 (v úseku I. P. Pavlova až Biskupcova). Jako další by se měl implementovat úsek I. P. Pavlova až Kotlářka popřípadě Sídliště Řepy, čímž by byla skoro dokončena linka a další značná část linek 10, 16 15. Tento úsek by také povolil demonstraci implementace mostů řek.



Obrázek 27: Plán přidání dalších tramvajových tratí, vyznačený na mapě PID.<sup>[40]</sup>

### 9.3. Vylepšení modelu osvětlení

Současně používá simulátor model osvětlení Blinn-Phong (viz kapitola 5.8 „Osvětlení“). budoucnu existuje možnost použití novějšího, avšak pomalejšího, modelu PBR, jinak známého jako Physically Based Rendering (fyzikálně založené vykreslování). Tento model, na rozdíl od modelu Blinn-Phong, zakládá svoje výpočty na fyzikálních jevech či zákonech, nikoliv na několika předem určených barvách konstantách.

### 9.4. Přidání řek mostů

Jakýkoliv další rozvoj projektu by požadoval podporu řek mostů přes nich. Řeky by bylo možné implementovat nastavením tzv. „úrovně vody“ na kterém by se vykreslovala rovina, popřípadě shaderem simulujícím efekt vody či ledu. Mosty by bylo potřeba nejdříve celé vymodelovat (a to včetně kolejí, pruhů apod.) následně napojit na dynamicky vygenerovanou síť tratí. Sloupy mostu by se taktéž museli prodloužit pod úroveň vody (jedná-li se most přes vodní tok) či pod úroveň terénu, aby nebylo jejich dno vidět. Malé potoky jsou důvodů časové obtížnosti zanedbány.

### 9.5. Přidání opravdového terénu

Současně jsou simulátoru implementovány pouze ulice, koleje budovy. Pro zvýšení realismu simulátoru by bylo potřeba simulovat opravdový okolní terén. Jeden způsob kterým by bylo možné tohoto docílit je přidání další vrstvy do scénáře která by určovala převýšení terénu daném místě. dostatečnou přesností terénu by bylo možné odebrat současně existující hodnoty výšky pro každý objekt odvozovat je od hodnot terénu.

### 9.6. Přidání nastavení trakčního vedení stahování pantografu

Jelikož na současně simulovaném úseku nejsou žádné mosty pod kterými by tramvaj projížděla kde by bylo potřeba snížení pantografu, není tato funkce simulátoru implementována. Současně je trakční vedení vykreslováno na určité výšce nad body tratě výška pantografu se nemění. Řešením tohoto problému pro budoucí rozvoj projektu je přidání další vrstvy do programu pro úpravy ve kterém by byly přidány výškové body pro trakční vedení, dle kterých by se snižoval zvyšoval pantograf. Je možné také přidat manuální snížení či zvýšení za pomoci lana kabině řidiče.

## **Závěr**

Projekt splnil valnou většinu původních očekávání. Obsahuje všechny zmíněné fyzikální simulace, světelnou zvukovou simulaci, to optimalizované na vlastním grafickém rozhraní. Samozřejmě, jak je zmíněno kapitole 9, projekt má ještě dalekou cestu do plného dokončení plné funkčnosti.

Tento simulátor je možné používat nejen jako nástroj pro trénování budoucí generace řidičů tramvají, ale také jako fyzikální demonstraci všech sil působících na kolejové vozidlo či jako společenskou demonstraci efektu používání kolejové MHD na emise skleníkových plynů.

Projekt mimo jiné také velice pomohl rozvoji dovedností týkajících se 3D modelování, konkrétně programu Blender, programování webových i desktopových aplikací, fyziky matematiky k tomu sestavování vlastních formátů implementace jejich ukládání načítání hned několika programovacích jazycích.

## Použité technologie

Práce byl zpracována za pomoci následujících technologií:

- Modelování bylo provedeno softwaru Blender.
- Infrastruktura byla implementována za pomoci jazyka C++ knihoven libc++, GLFW, GLAD, GLM, stbi, fastgltf, Dear ImGui miniaudio. Bylo taktéž použito grafické API OpenGL 4.5. Pro kompilování programu byl použit systém CMake.
- Webové nástroje byly implementovány za pomoci jazyka JavaScript technologií HTML CSS. Byla také použita knihovna Leaflet implementující embed map.
- Kreslení vymalování textur bylo provedeno softwaru Krita.
- Úprava zdrojových kódů byla provedena softwaru KDE Kate v softwaru Microsoft Visual Studio Code.
- Dokument prezentace byli sepsány programu Writer sady LibreOffice.
- Pro vykreslení grafů byla použita webová aplikace draw.io.
- Plakát pro práci byl vytvořen softwaru Inkscape.
- Jako operační systém byl využit Fedora GNU/Linux.

## Seznam obrázků

Obrázek 1: Fotografie vozů T3R.P (ev. č. 8234 8235) DPP nedaleko Sídliště Řepy <sup>[3]</sup> .....	14
Obrázek 2: Červeno-krémový nátěr na modelu tramvaje typu T3R.P.....	14
Obrázek 3: Vůz T3R.PLF (ev. č. 8283) vlečný vůz T3R.P (ev. č. 8160) na Palmovce <sup>[3]</sup> .....	14
Obrázek 4: Vínovo-stříbrný nátěr nanesený na modelu typu T3R.P.....	14
Obrázek 5: Umělecké zobrazení nátěru PID vytvořeného dle grafického manuálu PID.....	14
Obrázek 6: Proces vykreslování modelu grafickým rozhraním OpenGL 4.5.....	20
Obrázek 7: Obrázek popisující generaci tratí.....	24
Obrázek 8: Funkce, čísla vzhled jednotlivých tramvajových návěstí. <sup>[1][23][24]</sup> .....	26
Obrázek 9: Možné stavy výhybkových návěstidel tramvajovém simulátoru. <sup>[1]</sup> .....	28
Obrázek 10: Možné stavy křižovatkových návěstidel tramvajovém simulátoru. <sup>[1][24]</sup> .....	29
Obrázek 11: Fotografie nástroje pro úpravu mapy scénářů při jeho otevření.....	32
Obrázek 12: Fotografie režimu vytváření linek nástroji pro úpravu mapy scénářů.....	35
Obrázek 13: Fotografie menu nastavení linek nástroji pro úpravu mapy scénářů.....	35
Obrázek 14: Fotografie nástroje pro vytváření hlásiče při spuštění.....	43
Obrázek 15: Hodnoty použité při výpočtu síly nadzvednutí vozidla oblouku.....	49
Obrázek 16: Znázornění výpočtu difúzního světla dle (Blinn-)Phongova modelu.....	52
Obrázek 17: Znázornění výpočtu odraženého světla dle Blinn-Phongova modelu.....	52
Obrázek 18: Dopad světla pod úhlem bez aplikování posuvné hodnoty.....	54
Obrázek 19: Dopad světla pod úhlem po aplikování posuvné hodnoty (tzv. „bias“). ....	54
Obrázek 20: Spektrogram neupraveného opakujícího se zvuku.....	56
Obrázek 21: Spektrogram upraveného opakujícího se zvuku.....	56
Obrázek 22: Obrázek modelu zastávkového sloupu.....	61
Obrázek 23: Obrázek modelu rodinného domku.....	62
Obrázek 24: Obrázek modelu vícepatrového řadového domu.....	62
Obrázek 25: Obrázek modelu vícepatrového rohového řadového domu.....	63
Obrázek 26: Obrázek modelu panelového domu.....	63
Obrázek 27: Plán přidání dalších tramvajových tratí, vyznačený na mapě PID.....	66

## Seznam tabulek

Tabulka 1: Seznam barev použitých modelu.....	13
Tabulka 2: Konfigurační hodnoty pro typ T3R.P <sup>[13][14]</sup> CITACE ENGINEERINGTOOLBOX .....	16
Tabulka 3: Posunové hodnoty pro ovládání vozidla typu T3R.P.....	17
Tabulka 4: Další povinné konfigurační hodnoty.....	18
Tabulka 5: První číslice dle vozovny <sup>[1][27]</sup> .....	31
Tabulka 6: Seznam typů budov, jejich popisků rozměrů.....	39
Tabulka 7: Seznam památek, jejich popisků rozměrů.....	39
Tabulka 8: Efekt počasí různých stavů na koeficient tření.....	47
Tabulka 9: Hodnoty pro výpočet atenuace závislosti na dosahu světla.....	52
Tabulka 10: Barvy materiály budov.....	64
Tabulka 11: Seznam akcí za které se body odečítají.....	65
Tabulka 12: Seznam akcí za které se body přičítají.....	65

## Citace

Je-li citace označena hvězdičkou např. [1\*], vztahuje k celému předchozímu paragrafu. Není-li uvedena, vztahuje k předchozí větě, případně několika předchozím (Závisí na kontextu.). Citace jsou uvedeny dle normy ČSN ISO 690.

1. KASTNER, Eduard. ovládání tramvají. Praha, 23.1.2025 [osobní komunikace]
2. LEUBNER, Martin. tramvají, nahrávání zvuků na lince 91. Praha, 15.3.2025 [osobní komunikace]
3. SEZNAM-AUTOBUSU.CZ. seznam-autobusu.cz. Online. © 2007-2025. Dostupné z: <https://seznam-autobusu.cz/> [cit. 24. 11. 2024]
4. TRANSPORT ÎN COMUN ROMÂNIA. Tramvaie Tatra T3R.P. transport-in-comun.ro. Online. © 2006-2012. Dostupné z: [http://transport-in-comun.ro/romania/cehia/praga/tramvaie/t3r\\_p/t3r\\_p.html](http://transport-in-comun.ro/romania/cehia/praga/tramvaie/t3r_p/t3r_p.html) [cit. 24. 11. 2024]
5. DOPRAVNÍ PODNIK HL. M. PRAHY, AKCIOVÁ SPOLEČNOST, ODDĚLENÍ VÝCVIK VZDĚLÁVÁNÍ – TRAMVAJE. Tramvaje T3.SUCS, T3M, T3R.P, T3R.PLF. [Praha], 2009. Dostupné z: [https://www.prazsketramvaje.cz/obrazky/stahuj/testy/ucebnice\\_t3.pdf](https://www.prazsketramvaje.cz/obrazky/stahuj/testy/ucebnice_t3.pdf) [cit. 15. 3. 2025]
6. TECHNICKÉ MUZEUM BRNĚ, DEPOZIT VOZIDEL MHD. Brno, tramvaj T3 č. 521. Online. © 2002 Ivo Mahel (obrázek), 2025 (stránka informace). Dostupné z: <https://sabdigital.cz/expozicemhd/exponaty/tramvaje/brno-tramvaj-t3-c-521/> [cit. 14. 9. 2024]
7. TECHNICKÉ MUZEUM BRNĚ, ČKD PRAHA. Technický výkres Pantograf - výkres O-37-170059 – kopie. Online. © Brno [Praha], 2025 [1966]. Dostupné z: <https://sbirkytmb.museion.cz/?id=150091952> Webový portál: Brno, 2025, výkres Praha, 1966. [cit. 14. 9. 2024]
8. KLL-KONSTRUKČNÍ KANCELÁŘ, S.R.O. OSTRAVA. Modernizace tramvajového podvozku t3 sum - pro SKD TRADE, a.s. kll.cz. Online. [© 2008], [Ostrava]. Dostupné z: <http://www.kll.cz/pages/t3sumtramvajpodvozek.html> [cit. 19. 10. 2024]
9. DOPRAVNÍ PODNIK HL. M. PRAHY, A.S. DPP datech. dpp.cz. Online. © 2025. <https://www.dpp.cz/spolecnost/o-spolecnosti/dpp-v-datech>
10. MEZINÁRODNÍ ORGANIZACE PRO NORMALIZACI, WIKIPEDIA. wikipedia.org, iso.org. Dostupné z: [https://en.wikipedia.org/wiki/ISO\\_3864](https://en.wikipedia.org/wiki/ISO_3864) *Tento zdroj byl použit pouze jako zdroj barev použitých specifikaci ISO 3864.* [cit. 24. 11. 2024]

11. WIKIMEDIA COMMONS. Category:ISO 7010 safety signs. commons.wikimedia.org. Online. © 2023. Dostupné z: [https://commons.wikimedia.org/wiki/Category:ISO\\_7010\\_safety\\_signs](https://commons.wikimedia.org/wiki/Category:ISO_7010_safety_signs). [cit. 24. 11. 2024]
12. REGIONÁLNÍ ORGANIZÁTOR PRAŽSKÉ INTEGROVANÉ DOPRAVY, INTEGROVANÁ DOPRAVA STŘEDOČESKÉHO KRAJE, PRAŽSKÁ INTEGROVANÁ DOPRAVA. Manuál jednotného vzhledu vozidel PID, autobusy. pid.cz. Online. © 2023. Dostupné z: [https://pid.cz/wp-content/uploads/system/sk\\_bus/SK\\_bus\\_priloha\\_manual\\_jednotneho\\_vzhledu\\_vozidel\\_2024-01-01.pdf](https://pid.cz/wp-content/uploads/system/sk_bus/SK_bus_priloha_manual_jednotneho_vzhledu_vozidel_2024-01-01.pdf) [cit. 24.11.2024]
13. Dopravní podnik hl.m. Prahy, a.s. Tatra T3R.P. dpp.cz. Online. © 2025. Dostupné z: [www.dpp.cz/vozovy-park/tramvaje/tatra-t3r-p](http://www.dpp.cz/vozovy-park/tramvaje/tatra-t3r-p) [cit. 11. 3. 2025]
14. MOBILNÍ TABLA. Informace tramvaji 8228. tram.mobilnidata.cz. Online. © 2022. Dostupné z: <https://tram.mobilnitabla.cz/vuz/8228> [cit. 19. 3. 2025]
15. CPPREFERENCE.COM. C++ reference. en.cppreference.com. Online. © 2025. Dostupné z: <https://en.cppreference.com/w/> [cit. 19. 3. 2025]
16. PETERSEN, Christian. ANSI Escape Sequences. gist.github.com. Online. © 2025. Dostupné z: <https://gist.github.com/fnky/458719343aabd01cfb17a3a4f7296797> [cit. 17. 3. 2025]
17. KHRONOS GROUP, Rendering Pipeline Overview. khronos.org. Online. © 2022. Dostupné z: [https://www.khronos.org/opengl/wiki/Rendering\\_Pipeline\\_Overview](https://www.khronos.org/opengl/wiki/Rendering_Pipeline_Overview) [cit. 17. 3. 2025]
18. KHRONOS GROUP, Interface Block (GLSL): Memory Layout. khronos.org. Online. (c) 2022. Dostupné z: [https://www.khronos.org/opengl/wiki/Interface\\_Block\\_\(GLSL\)#Memory\\_layout](https://www.khronos.org/opengl/wiki/Interface_Block_(GLSL)#Memory_layout) [cit. 17. 3. 2025]
19. KHRONOS GROUP. Shader Compilation. khronos.org. Online. © 2022. Dostupné z: [https://www.khronos.org/opengl/wiki/Shader\\_Compilation](https://www.khronos.org/opengl/wiki/Shader_Compilation) [cit. 17. 3. 2025]
20. de VRIES, Joey. Learn OpenGL – Graphics Programming: Learn modern OpenGL graphics programming in step-by-step fashion. 1. vydání. Kendall&Welling. 2020. ISBN 978-90-90-33256-7 Dostupné online ve formátu PDF z: [https://learnopengl.com/book/book\\_pdf.pdf](https://learnopengl.com/book/book_pdf.pdf). Zdroj také dostupný ve webové podobě na adrese <https://learnopengl.com>. [cit. 27. 2. 2025]
21. HESHMATI MOGHADDAM, Mahan. LearnOpenGL: OIT: Weighted Blended. learnopengl.com. Online. © 2025. Dostupné <https://learnopengl.com/Guest-Articles/2020/OIT/Weighted-Blended>. [cit. 24.2.2025]
22. DEMES, Lennart. AmbientCG – free textures, HDRIs & More. ambientcg.com. Online. Dostupné z: <https://ambientcg.com/> [cit 15. 3. 2025]



23. PRAŽSKÉ TRAMVAJE. Návěsti návěstidla tramvajové dopravě. prazsketramvaje.cz. Online. © 2001-2025. ISSN 1801-9994. Dostupné z: <https://www.prazsketramvaje.cz/view.php?cisloclanku=2006041260> [cit. 16. 1. 2025]
24. PRAŽSKÉ TRAMVAJE. Návěsti upravující rychlost tramvaje. prazsketramvaje.cz. Online. © 2001-2025. ISSN 1801-9994. Dostupné z: <https://www.prazsketramvaje.cz/view.php?cisloclanku=2006041265> [cit. 16. 1. 2025]
25. PRAŽSKÉ TRAMVAJE. Světelné návěsti pro tramvaje. prazsketramvaje.cz. Online. © 2001-2025. ISSN 1801-9994. Dostupné z: <https://www.prazsketramvaje.cz/view.php?cisloclanku=2006041263> [cit. 16. 1. 2025]
26. CHODOUNSKÝ, Josef. Pořadí na lince. rezistoruv.blogspot.com. Online, blogový příspěvek. © 2019. Dostupné z: <https://rezistoruv.blogspot.com/2020/05/poradi-na-lince.html> [cit. 16. 3. 2025]
27. MOZILLA WEB DOCS. Web technology for developers. developer.mozilla.org. Online. © 2025. Dostupné z: <https://developer.mozilla.org/en-US/docs/Web> [cit. 7. 1. 2025]
28. AGAFONKIN, Volodymyr kol. Leaflet API reference. leafletjs.com. Online. © 2025. Dostupné z: <https://leafletjs.com/reference.html> [cit. 7. 1. 2025]
29. ENGINEERINGTOOLBOX. Rolling Resistance. engineeringtoolbox.com. Online. © 2008. Dostupné z: [https://www.engineeringtoolbox.com/rolling-friction-resistance-d\\_1303.html](https://www.engineeringtoolbox.com/rolling-friction-resistance-d_1303.html) [cit. 24. 3. 2025]
30. HARMON, M.; SANTA, J.F.; JARAMILLO, J.A. kol. Evaluation of the coefficient of friction of rail in the field and laboratory using several devices. Online. © 2020. Dostupné z: <https://eprints.whiterose.ac.uk/156304/3/Newton%20Friction%20Journal%20Paper%20V2.pdf> [cit. 16. 3. 2025]
31. VOLTR, Petr; MLYNÁŘÍK, Ladislav; NOVÁK, Jaroslav; MICHÁLEK, Tomáš. Rozjezd vlaku okem studenta železniční techniky na DFJP. Online, příspěvek. © 2015. Dostupné z: <https://www.zelpage.cz/clanky/rozjezd-vlaku-okem-studenta-zeleznici-techniky-na-dfjp>. [cit. 24. 3. 2025]
32. GHOSH, Shibashis. Tractive Force Calculation for Vehicle. mechguru.com. Online. © 2013. Dostupné z: <https://mechguru.com/vehicle-design/tractive-force-calculation-for-a-vehicle/> [cit. 24. 3. 2025]
33. MARQUIS, Brian; GREIF, Robert. Application of nadal limit in the prediction of wheel climb derailment. In: ASME/IEEE Joint Rail Conference. 2011. p. 273-280. Dostupné z: [https://www.railwayage.com/wp-content/uploads/2020/12/JRC2011-56064\\_nadal.pdf](https://www.railwayage.com/wp-content/uploads/2020/12/JRC2011-56064_nadal.pdf) [cit. 16. 3. 2025]

34. ENGINEERINGTOOLBOX. Drag Coefficient. engineeringtoolbox.com. Online. © 2004. Dostupné z: [https://www.engineeringtoolbox.com/drag-coefficient-d\\_627.html](https://www.engineeringtoolbox.com/drag-coefficient-d_627.html) [cit 25. 1. 2025]
35. SIDNEY, FMOTION1. What circumstances or material types should have different specular value in PBR workflow? blender.stackexchange.com. Online, příspěvek. © 2021. Dostupné z: <https://blender.stackexchange.com/a/234200> [cit. 15. 12. 2024]
36. UNITED STATES ENVIRONMENTAL PROTECTION AGENCY. Greenhouse Gas Emissions from Typical Passenger Vehicle. epa.gov. Online. © 2024. Dostupné z: <https://www.epa.gov/greenvehicles/greenhouse-gas-emissions-typical-passenger-vehicle> [cit. 24. 3. 2025]
37. EMBER; ENERGY INSTITUTE; OUR WORLD IN DATA. Carbon intensity of electricity generation. ourworldindata.org. Online. © 2024. Dostupné z: <https://ourworldindata.org/grapher/carbon-intensity-electricity> [cit. 24. 3. 2025]
38. GŁÓWNY URZĄD STATYSTYCZNY. Energia 2024 - Folder zawierający podstawowe informacje produkcji wykorzystaniu paliw, ciepła energii elektrycznej 2023 r. Polsce. stat.gov.pl. Online. © 2024. Dostupné z: <https://stat.gov.pl/obszary-tematyczne/srodowisko-energia/energia/energia-2024,1,12.html> [cit. 24. 3. 2025]
39. ČESKÁ NÁRODNÍ BANKA. Kurzy devizového trhu. cnb.cz. Online. © 2025. Dostupné z: [https://www.cnb.cz/cs/financni-trhy/devizovy-trh/kurzy-devizoveho-trhu/kurzy-devizoveho-trhu/denni\\_kurz.txt?date=21.03.2025](https://www.cnb.cz/cs/financni-trhy/devizovy-trh/kurzy-devizoveho-trhu/kurzy-devizoveho-trhu/denni_kurz.txt?date=21.03.2025) [cit. 24. 3. 2025, data 21. 3. 2025]
40. MACKŮ Pavel, PRAŽSKÁ INTEGROVANÁ DOPRAVA. Linky tramvají, trvalý stav platný od 25. 11. 2024. pid.cz. Online. © 2024. Dostupné z: [https://pid.cz/wp-content/uploads/mapy/schemata-trvala/a4\\_tramvajovy\\_provoz.pdf](https://pid.cz/wp-content/uploads/mapy/schemata-trvala/a4_tramvajovy_provoz.pdf) [cit. 27. 12. 2024]

Následující zdroje byli použity při psaní programu, nejsou avšak samotné dokumentaci práce zmíněny.

41. APELER, Sean. FastglTF Documentation. readthedocs.io. Online. © 2025. Dostupné z: <https://fastgltf.readthedocs.io/v0.8.x/index.html> [cit. 25. 2. 2025]
42. APELER, Sean. gl\_viewer.cpp. github.com. Online. © 2025. Dostupné z: [https://github.com/spnda/fastgltf/blob/main/examples/gl\\_viewer/gl\\_viewer.cpp](https://github.com/spnda/fastgltf/blob/main/examples/gl_viewer/gl_viewer.cpp) [cit. 1. 2. 2025]
43. RICCIO, Christophe. OpenGL Mathematics library. Online. © 2025. Dostupné z: <https://glm.g-truc.net/0.9.4/api/> [cit. 22. 2. 2025]
44. VON GAGERN, Martin. Direct way of computing the clockwise angle between two vectors. stackoverflow.com. Online, příspěvek. © 2013. Dostupné z: <https://stackoverflow.com/a/16544330> [cit. 22. 2. 2025]

45. ANONYMNÍ UŽIVATEL [meowgoethedog], How to get right Up vector for LookAt transformation? stackoverflow.com. Online, příspěvek. © 2017. Dostupné z: <https://stackoverflow.com/a/47188758> [cit. 9. 3. 2025]
46. GLFW. Documentation. glfw.org. Online. © 2025. Dostupné z: <https://www.glfw.org/docs/latest/> [cit. 20. 3. 2025]
47. REES, Gareth. How do you detect where two line segments intersect? stackoverflow.com. Online, příspěvek. © 2009. <https://stackoverflow.com/a/565282> [cit. 22. 3. 2025]
48. OPENGL-TUTORIAL. Picking with custom Ray-OBB function. opengl-tutorial.org. Online, blogový příspěvek. © 2013. Dostupné z: <https://www.opengl-tutorial.org/miscellaneous/clicking-on-objects/picking-with-custom-ray-obb-function/> [cit. 15. 3. 2025]
49. OPENGL-TUTORIAL. Billboards: The 3D Way. opengl-tutorial.org. Online, blogový příspěvek. © 2013. Dostupné z: <https://www.opengl-tutorial.org/intermediate-tutorials/billboards-particles/billboards/#solution-2--the-3d-way> [cit. 9. 3. 2025]
50. BITTLE, William. SAT (Separating Axis Theorem). dyn4j.org. Online, blogový příspěvek. © 2010. Dostupné z: <https://dyn4j.org/2010/01/sat/> [cit. 11. 3. 2025]
51. ANONYMNÍ UŽIVATEL [david]. glm translate matrix does not translate the vector. stackoverflow.com. Online, příspěvek. © 2013. Dostupné z: <https://stackoverflow.com/a/18988838> [cit. 13. 3. 2025]
52. BAHANDOV, Dmytro. Ray Oriented bounding box (OBB) intersection function not working with scale. stackoverflow.com. <https://stackoverflow.com/a/76062770> [cit. 16. 3. 2025]
53. REID, David. Miniaudio Programming Manual. miniaud.io. Online. © 2025. Dostupné z: <https://miniaud.io/docs/manual/index.html> [cit. 24. 3. 2025]