Středoškolská technika 2014

Setkání a prezentace prací středoškolských studentů na ČVUT

Animace pohybu vozidla

Martin Kotík

Střední průmyslová škola na Proseku
Novoborská 2, 190 00 Praha 9

1 Obsah

1 Obsah 1
2 Část 1. 3D grafika (základní pojmy) 3
2.1 3D Modely 3
3 Typy 3D modelů 3
3.1 NURBS povrch 3
3.2 Polygonální Model 3
3.2.1 Plochy 4
3.2.2 Hrany 4
3.2.3 Vrcholy 4
3.2.4 Textury a shadery 4
3.3 Shader 4
3.4 Textury 5
4 Scéna 5

5 Rendering 5
6 Kartézská soustava souřadnic 6
7 Animace 6
7.1 Key frame 7
7.2 Inbetweening 7
8 Softwarové nástroje pro 3d grafiku 7
9 Software pro 3D modeling 7
9.1 Cinema 4D 7
9.1.1 Moduly 8
9.1.2 Cinebench 8
9.1.3 3D renderer 8
9.1.4 CAD software 8
10 specifika hardware pro 3D grafiku 9
11 Procesor (CPU) 9
11.1 Více vláknový výpočetní výkon 9
11.2 Jedno vláknový výpočetní výkon 9
12 Grafická karta (GPU) 9
12.1 Výkon v editoru 9
12.2 Profesionální řešení – grafické karty 9
12.3 Technologie grafických karet 10
12.3.1 OpenGL 10
12.3.2 OpenCL 10
12.3.3 CUDA 10
13 Modelování Pomocí blueprintu 10
13.1 Blueprint 11
14 Popis Práce 13
15 Seznam použitých zdrojů 26

2 Část 1. 3D grafika (základní pojmy)

3D grafika je grafika, která využívá tří dimenzionální (prostorové) zobrazení geometrických dat (viz. Kartézská soustava souřadnic). Spoléhá na mnoho stejných algoritmů jako 2D vektorová počítačová grafika ve wireframe modelu a 2D rastrová grafika v konečném zobrazování. 2D aplikace mohou používat 3D techniky k dosažení požadovaných efektů, jako je osvětlení a 3D mohou používat 2D renderovací techniky.

2.1 3D Modely

Části 3D počítačové grafiky jsou často nazývány jako 3D modely (objekty). Model je základním stavebním prvkem 3D grafiky. 3D model je matematická reprezentace nějakého trojrozměrného objektu (skutečného či imaginárního) ve 3D softwarovém prostředí. Na rozdíl od 2D obrazu, 3D modely se dají prohlížet ve specializovaných programech z jakéhokoliv úhlu, a mohou být zmenšeny, otáčeny, nebo volně upravovány. Proces vytváření a formování 3D modelu je známý jako 3D modelování. Díky 3D tisku nejsou 3D modely omezeny pouze na virtuální prostor. Model může být zobrazen vizuálně jako dvourozměrný obraz prostřednictvím procesu nazývaného renderování, nebo použit v negrafické počítačové simulaci a výpočtech.

3 Typy 3D modelů

Existují dva základní typy 3D modelů, které jsou použity ve filmu či herním průmyslu, nejvíce zřejmých rozdílů je ve způsobu, jakým jsou vytvořeny a manipulovány (jsou mezi nimi také rozdíly v základní matematice, ale to je méně důležité pro koncového uživatele).

3.1 NURBS povrch

NURBS povrch je hladký povrch modelu vytvořen pomocí Bézierových křivek (v podstatě 3D verze MS Paint nástrojem pero). Pro vytvoření NURBS plochy, umělec kreslí dvě nebo více křivek v 3D prostoru, kterými lze manipulovat pomocí pohyblivých rukojetí (tzv. řídící vrcholy) podél X, Y, nebo Z osy.

Softwarová aplikace interpoluje prostor mezi křivkami a vytváří mezi nimi hladký mesh. NURBS plochy mají nejvyšší úroveň matematické přesnosti, a proto se nejčastěji používá při modelování pro inženýrství a automobilový design či design obecně.

3.2 Polygonální Model

polygonální modely nebo "mesh (meshes)", jak jsou často nazývány, jsou nejčastější formou 3D modelu, které se dají naleznout v animaci, filmu či herním průmyslu.

Polygonální modely jsou velmi podobné geometrickým tvarům. Stejně jako základní geometrické krychle, kvádry, jehlany, 3D polygonální modely se skládají z ploch, hran a vrcholů.

Ve skutečnosti, většina složitých 3D modelů začíná jako jednoduchý geometrický tvar, jako krychle, koule, či válce. Tyto základní 3D tvary se nazývají primitiva. Primitiva pak lze modelovat a tvarovat na jakýkoliv objekt, který se umělec snaží vytvořit.

3.2.1 Plochy

Charakteristickým znakem polygonálního modelu je to, že (na rozdíl od NURBS plášťů), polygonální mřížky jsou „faceted“, což znamená, že povrch 3D modelu se skládá ze stovek nebo tisíců geometrických ploch.

V modelování, polygony jsou buď čtyřhranné (normou v modelování postav/ organickém modelování) nebo tříhranné (používá častěji ve hrách). Umělci musí usilovat o efektivitu a organizaci tzn., snaží se udržet počet polygonů potřebný pro dosažení výsledku co nejnižší.

Počet polygonů v meshi, se nazývá „poly-count“, zatímco hustota polygonů, se nazývá rozlišení. Nejlepší 3D modely mají vysoké rozlišení. Tam kde je vyžadováno, jako například ruce či obličej postavy. V podstatě, čím vyšší je celkové rozlišení modelu, tím hladší se objeví v konečném renderu. Model s nižším rozlišením vypadá „krabicovitě“(např. staré hry).

3.2.2 Hrany

Hrana je kterýkoliv bod na povrchu 3D modelu, kde se dvě polygonální plochy spojují.

3.2.3 Vrcholy

Průsečíky mezi třemi nebo více hranami se nazývají vrcholy (j. Vertex mn. Vertices). Manipulace s vrcholy na X, Y a Z osách je nejčastější technikou pro tvarování polygonálního meshe do jeho finální podoby v tradičních modelovacích programy jako je Cinema4D, Maya, 3ds Max, atd. (techniky jsou velmi odlišné od tvarování (sculpting) v aplikacích, jako je ZBrush nebo Mudbox).

3.2.4 Textury a shadery

Ačkoli textury a shadery nemají nic do společného s celkovým tvarem 3D modelu, mají co do činění s jeho vzhledem.

3.3 Shader

shader je sada instrukcí, použitých na 3D model, které umožňují počítači vědět, jak má být zobrazen. I když mohou stínové sítě (shading networks) být kódovány ručně, většina 3D modelovacích programů má nástroje, pomocí kterých umožňují umělci jednoduše upravit parametry shaderu. Díky těmto nástrojům, může umělec kontrolovat interakci, povrchu modelu se světlem, včetně průhlednosti, odrazivosti, lesku, a desítky dalších.

3.4 Textury

Textury rovněž významně přispívají k vizuálnímu vzhledu modelu. Textury jsou dvourozměrné obrazové soubory, které mohou být mapovány na povrchu 3D modelu přes proces známý jako mapování textur. Textury se mohou lišit ve složitosti od jednoduchých barevných textur, až po zcela fotorealistické povrchové detaily.

4 Scéna

Jedná se o základní 3D plochu, vše vytvořené se na ní odehrává. Ve scéně je možné pohybovat se všemi směry, nahoru, dolu, doprava, doleva, dopředu i dozadu, ale také libovolně se otáčet. Jsou v ní uložené všechny souřadné systémy objektů a v případě natáčení scény se jednotlivé souřadné systémy objektů převádí na globální souřadný systém, ve kterém je otáčí a následně je zpětně převádí do původního souřadného systému.

5 Rendering

Česky vykreslování. Rendering je proces generování obrazu z modelu (nebo modelů v tom, co se společně dá nazvat scéna), pomocí počítačových programů. Výsledky tohoto procesu se nazývají render. Scéna obsahuje objekty v přesně vymezené datové struktuře, bude obsahovat geometrii zdroj pohledu (viewpoint), textury, osvětlení, informace o stínech jako popis virtuální scény. Údaje obsažené v souboru scény jsou pak předány renderovacímu programu ke zpracování a výstupu na digitální obraz nebo obraz rastrové grafiky.

Termín „vykreslování“, může být přirovnán k uměleckému ztvárnění scény. I když se technické detaily vykreslovacích metod liší, obecný postup zůstává stejný, výroba 2D obrazu z 3D reprezentace scény.

Rendering je také používán k popisu procesu výpočtu efektů v programu pro střih videa pro produkci finálního video výstupu.

Rendering je jedním z hlavních dílčích témat 3D počítačové grafiky a v praxi je vždy spojen s ostatními. V grafickém postupu, je posledním významným krokem, takže udává konečný vzhled modelů a animací. S rostoucí propracovanosti počítačové grafiky od roku 1970, se stal mnohem výraznějším předmětem.

Rendering má využití v architektuře, videohrách, simulátorech, filmech nebo televizních vizuálních efektů a vizualizacích návrhů, z nichž každé využívají jinou rovnováhu funkcí a technik. Jako produktu, je široká škála renderovacích programů k dispozici. Některé z nich jsou integrovány do větších modelovacích a animačních balíčků (Cinema 4D), některé jsou samostatné. Některé jsou zdarma open-source projekty. Uvnitř je renderer pečlivě navržený program založený na selektivní směsi oborů souvisejících s: světelnou fyzikou, zrakovým vnímáním, matematikou a vývojem software.

V případě 3D grafiky, rendering může být proveden pomalu, jak v pre-renderingu, nebo v reálném čase. Pre-rendering je výpočetně náročný proces, který se obvykle používá pro vytvoření filmu, zatímco real-time

rendering je často dělán pro 3D video hry, které jsou založeny na využití grafických karet s 3D hardwarovými akcelerátory.

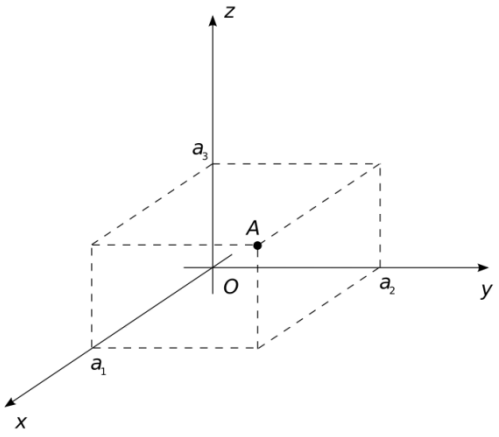
6 Kartézská soustava souřadnic

Kartézský souřadný systém je souřadnicový systém, který určuje bod unikátně v rovině dvojicí číselných souřadnic, které určují vzdálenost bodu od dvou na sebe kolmých protínajících se linek, měřených ve stejné jednotce délky. Každá referenční čára se nazývá osa systému a místo, kde se osy setkají je jejich původ. Obvykle uspořádaná trojce (X, Y).

Stejný princip se dá použít pro určení polohy libovolného bodu v trojrozměrném prostoru pomocí tří kartézských souřadnic, jeho vzdálenost ke třem vzájemně kolmým rovinám. (X,Y,Z)

O vynalezení kartézských souřadnic se v 17. století postaral René Descartés a zajistil revoluci v matematice tím, že vytvořil systematický vztah mezi euklidovskou geometrií a algebrou. Geometrické tvary, jako jsou křivky, mohou být popsány kartézskými rovnicemi.

Kartézské souřadnice jsou základem analytické geometrie, a poskytují poučné geometrické interpretace mnoha dalším odvětvím matematiky, jako je lineární algebra, komplexní analýza, diferenciální geometrie a další.



Kartézské souřadnice jsou také základní nástroje pro většinu aplikovaných disciplín, které se zabývají geometrií, včetně astronomie, fyzik, strojírenství a tak dále. Jedná se o nejběžnější souřadnicový systém používaný v počítačové grafice, geometrickém designu, a dalším zpracování dat související s geometrií.

7 Animace

Je pohyb a umístění objektů v rámci scény. Pod pojmem „animace“ se ve 3D grafice nerozumí pouze samotný pohyb objektů, ale i definice zdrojů světla, úhlu pohledu kamery, barev a dalších prvků, které se mohou měnit v čase. Nejjednodušší metoda animace zvaná keyframing je založená na stejném principu jako klasická 2D počítačová animace. Spočívá v definování klíčových „mezních“ pozic, mezi kterými potom počítač vytvoří plynulý přechod.

7.1 Key frame

(klíčový snímek) - Klíčový snímek v animaci či filmu je výkres, který definuje počáteční a koncové body každém hladkém přechodu. Výkresy jsou nazývány „frames", protože jejich pozice v čase se měří ve framech na pásu filmu. Sekvence klíčových snímků, které definují, který pohyb divák uvidí, zatímco pozice klíčových snímků ve filmu, videu či animaci definuje načasování pohybu. Vzhledem k tomu, že jen dva nebo tři klíčové snímky přes rozpětí sekundy nevytvářejí iluzi pohybu, zbývající frames jsou vyplněny „inbetweens“.

7.2 Inbetweening

Inbetweening nebo také tweening je proces generování mezilehlých snímků mezi dvěma obrázky tak, aby dojem, že první obraz se vyvíjí hladce do druhého. „Inbetweens“ jsou výkresy mezi klíčovými snímky, které pomáhají vytvořit iluzi pohybu. Inbetweening je klíčový proces ve všech typech animace, včetně počítačové animace.

8 Softwarové nástroje pro 3d grafiku

Současný trh je plný, plný softwarových nástrojů, které složí k vývoji 3D grafiky. Dělí se na tři hlavní skupiny, Software pro 3D modeling, 3D Renderery, CAD software.

9 Software pro 3D modeling

Je třída software počítačové grafiky používaný k výrobě 3D modelů. Programy této třídy se nazývají aplikace pro modelování (angl. Modelers). Programy umožňují uživateli vytvořit a měnit modely prostřednictvím 3D meshe. Uživatel může libovolně měnit mesh ke svému obrazu. Na modely může být pohlíženo z různých úhlů a vzdáleností. Aplikace mohou exportovat modely do souborů, které pak mohou být použity v jiných aplikacích, pokud jsou jejich metadata kompatibilní. Většina programů může číst data souborů ve formátu jejich původních aplikací. Hodně Aplikací obsahují spoustu funkcí, které úzce souvisí s modelováním, například umožňují animaci modelů či rozšířené možnosti renderování.

9.1 Cinema 4D

Cinema 4D je program vytvořený pro 3D modelování, animaci a rendering. Vyvinutý německou firmou MAXON Computer GmbH.

V současné době existují 4 varianty produktu: Základní Cinema 4D Prime. Broadcast, který přidává schopnosti motion-graphics. Visualize, který umožňuje funkce pro použití materiálů a textur a Studio, které obsahuje všechny předešlé moduly a navíc obsahuje například možnost používání funkcí Dynamics.

9.1.1 Moduly

Do verze R11.5 (současná verze je R15) měl uživatel možnost rozšiřovat (dokupovat) základní jádro aplikace o různé moduly. To se změnilo s příchodem verze R12 kdy se různé moduly rozdělily do čtyř verzí Cinemy 4D.

Advanced Render - je modul pro pokročilé nastavení renderingu (HDRI, Caustic, Ambient Oclussion, Global Ilumination). Modul tak umožňuje dosažení mnohem realističtějšího vzhledu výsledného renderu.

BodyPaint 3D - modul pro pokročilé texturování a malování přímo na 3D objekty

Dynamics - modul pro simulaci dynamiky měkkých a tvrdých těles a fyzikálních zákonů

Hair - modul pro tvorbu vlasů, chlupů, srsti a trávy

MOCCA - modul pro animaci postav, možnost animace oděvů

MoGraph - modul pro tvorbu televizní grafiky a reklam

NET Render - modul umožňující rendering po síti - spojení výpočetního výkonu několika počítačů

PyroCluster - modul pro snadnou tvorbu ohně, kouře. Od verze modulu Advanced Render 2.6 se PyroCluster stal jeho součástí.

Sketch and Toon - modul pro tvorbu kresleného vzhledu výsledného obrázku

Thinking Particles - modul pro pokročilou tvorbu částicových systémů

Xpresso - (není a nikdy nebyl modul, je to základní funkce).

9.1.2 Cinebench

Cinema 4D má též svůj vlastní bezplatný nástroj pro benchmarking ve Windows a Mac OS nazývaný Cinebench. Výsledky testů provedených pomocí programu nesou velkou váhu při analýze počítačového výkonu pro každodenní používání. Hlavním cílem je testování výkonu CPU a GPU (OpenGL), Cinebench je multiplatformní tudíž nezáleží na druhu operačního systému, díky tomu může být porovnáván například výkon PC oproti MAC.

9.1.3 3D renderer

I když většina 3D modelovacích programů či CAD software, umí provést 3D rendering (Cinema 4D, Autodesk Maya, Autodesk 3ds Max, Blender), existují i specializované programy přímo pro 3D rendering, například: Vray, Arnold, Maxwell.

9.1.4 CAD software

Computer aided design obsahuje základní modelovací funkce podobné 3D modelérům, ale jejich účel se velmi liší. Jsou totiž používány pro počítačově podporované inženýrství (CAE), počítačově podporované výroby (CAM), modely pro 3D tisk. Například: Autodesk Inventor, NX CAE.

10 specifika hardware pro 3D grafiku

Software pro modelování a 3D grafiku obecně je velmi náročný na výkon počítače, to znamená, že výběr správného hardware je pro bezproblémovou práci se softwarem zásadní.

11 Procesor (CPU)

Procesor je velmi důležitou komponentou, jeho výkon ovlivňuje zejména při výsledném výpočtu vytvořených dat, ale také na rychlost editoru, protože je na něm závislý výkon GPU, který editor zobrazuje.

11.1 Více vláknový výpočetní výkon

Je zapotřebí při výpočtech scény například v Cinema 4D. Čím vyšší výkon tím rychleji se obraz scény spočítá. Většinou platí, že Fyzikální engine lepé využívá možnosti více jader než engine standartní.

11.2 Jedno vláknový výpočetní výkon

Je potřebný pro zpracovávání dat přímo v editoru vyslaných grafické kartě k zobrazení. Nízký výkon přímo ovlivňuje plynulost práce v editoru i přesto, že výkon grafické karty je vysoký. Jedno vlákno se také uplatňuje na před výpočty některých úloh, ale i na exportu videa.

12 Grafická karta (GPU)

Výkon grafického procesoru můžeme opět rozdělit na dvě části. První část je rychlost zobrazení práce v editoru. Grafická karta se ale dá použít i v jistých typech výpočtů, v takovém případě se jedná o výpočty probíhající v jiných systémech, často implementovaných přímo v programu, které ale nejsou jejich přímou součástí.

12.1 Výkon v editoru

Výkon je hlavně závislý na výkonu CPU (jak již bylo řečeno). Čím je výkon větší, tím rychleji se zpracovávají data pro grafickou kartu a tudíž se zvyšuje rychlost (plynulost) editoru. CPU, ovlivňuje i data, které jsou nejdříve poslány k výpočtu a poté k zobrazení. Například modely Nurbs či složíte deformátory. V tomto případě rychlost editoru zatěžuje více výkon CPU než GPU.

Obecně platí, že by grafické karty měly mít vlastní paměť RAM, která není sdílená s RAM počítače, jak tomu je například u mobilních zařízení.

12.2 Profesionální řešení – grafické karty

Oproti herním grafickým kartám mají karty specializované na výpočet grafických dat nižší takt procesoru, tudíž vyšší životnost a potenciálně vyšší možnosti RAM. Podporují širší možnosti výstupů, například

display port či podporu více zobrazovacích panelů najednou a také podporu vyspělejších technologií. Další výraznou výhodou jsou certifikované ovladače na širokou škálu aplikací, čímž zajišťují jejich funkčnost. Hlavní nevýhodou těchto karet je vysoká pořizovací cena, která se může pohybovat i v řádech desetitisíců. (příklad: Nvidia Quadro, Nvidia Tesla, AMD FirePro)

12.3 Technologie grafických karet

12.3.1 OpenGL

Je technologie, multi platformní API (application programming interface), která je používána pro renderování 2D a 3D grafiky. Uplatňována většinou prostřednictvím grafických karet. Hlavní uplatnění má v zobrazování grafiky v reálném čase (vizualizace 3D grafiky, počítačových her, průmyslových simulací a animací).

12.3.2 OpenCL

Je otevřené programovací prostředí (framework) pro vývoj aplikací, běžících v prostředí heterogenních výpočetních systémů, které zahrnují procesory, grafické karty, specializované procesorové systémy apod.

Hlavní výhodou OpenCL je podpora paralelizovaných výpočtů, což je například rendering v 3D grafických programech. Běh jeho aplikací je přímo závislý na podpoře výrobců, tudíž na existenci ovladačů. Jeho alternativou je systém

12.3.3 CUDA

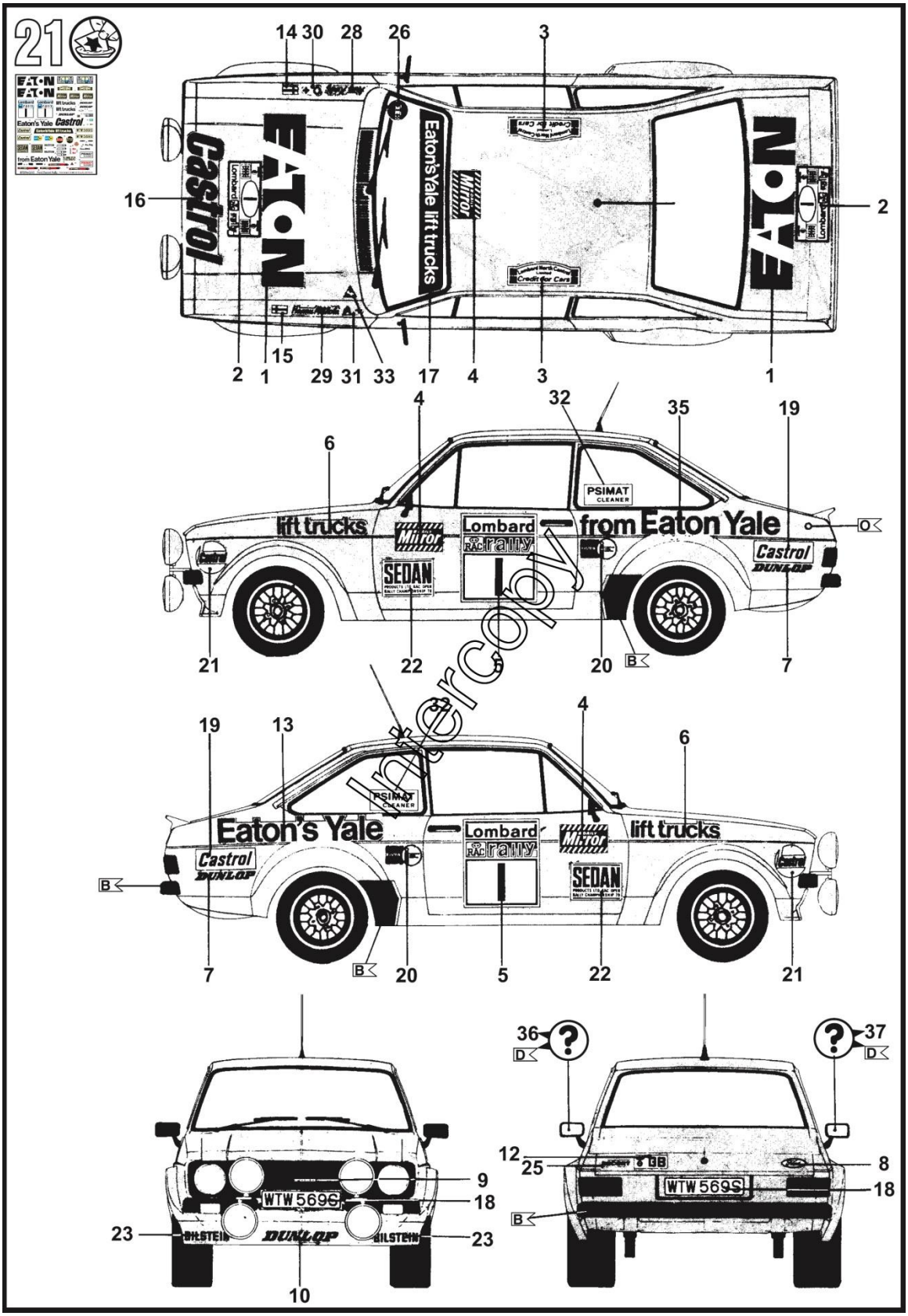
Je platforma od společnosti Nvidia pro vývoj paralelizovaných aplikací. Běžících na jejich grafických kartách. Pro tvorbu aplikací na platformě CUDA jsou používány programovací jazyky C/C++ či Fortran. V současné době je díky vyššímu výkonu a dřívějšímu příchodu na trh CUDA více podporována výrobci software než OpenCL.

13 Modelování Pomocí blueprintu

Modelování pomocí blueprintu je jedním z nejjednodušších způsobů jak modelovat složitější objekty(auta, stroje obecně a dokonce i postavy) a to i proto, že existují tisíce takových blueprintů, které je možné si zdarma stáhnout a použít. Díky blueprintu lze oproti klasickému modelování „naprázdno“ docílit extrémní přesnosti a hlavním důvodem pro to je, že si uživatel nemusí dělat starosti s tvary, vzdálenostmi či úhly, protože vše už je dané na pozadí, kterým se řídí. I přesto, že je blueprint při modelování velmi nápomocný, nedokáže kvůli pouze čtyřem pohledům zobrazit všechny detaily objektu, a proto musí uživatel současně používat obrázky objektu (třeba z internetu) z různých úhlů jako referenci. Drtivá většina Grafických 3D programů podporuje funkci užívání blueprintu. Nejlépe je využit pro point by point modeling či spline modeling.

13.1 Blueprint

je reprodukce technického výkresu, která dokumentuje architekturu či konstrukční návrh, použitím tiskového procesu na světlo citlivé listy.

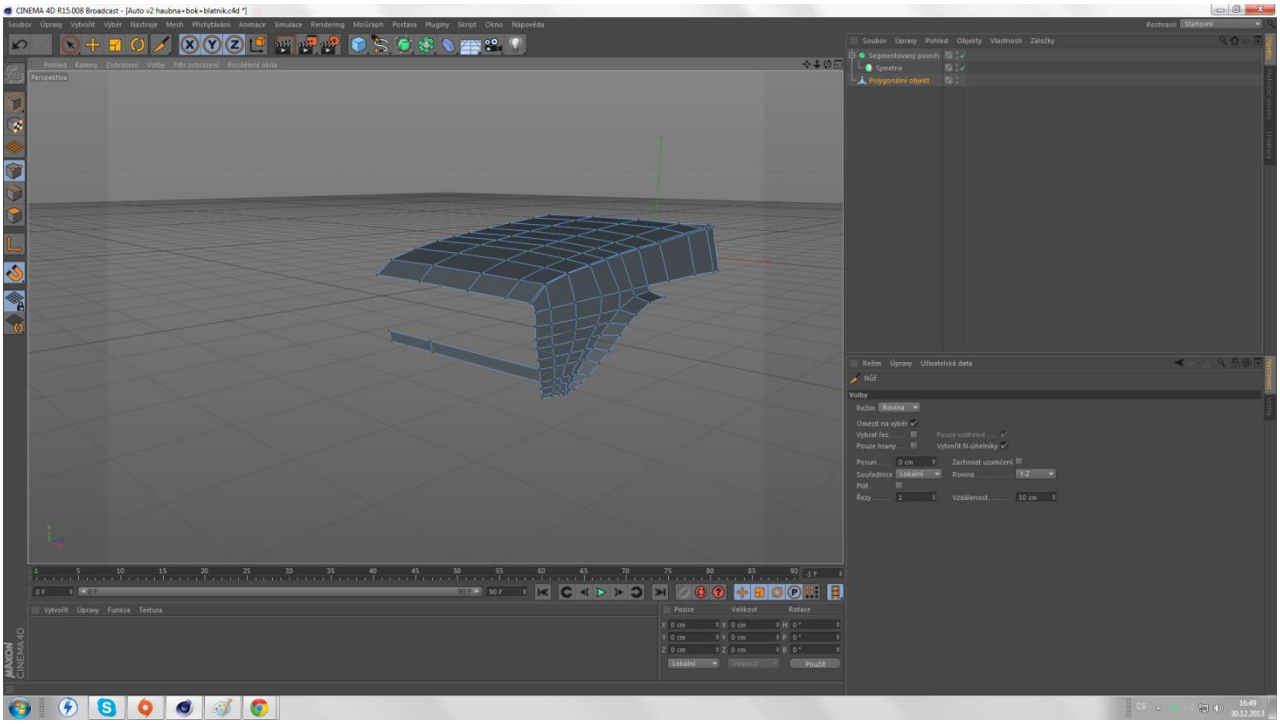


Ukázka blueprintu použitého v projektu

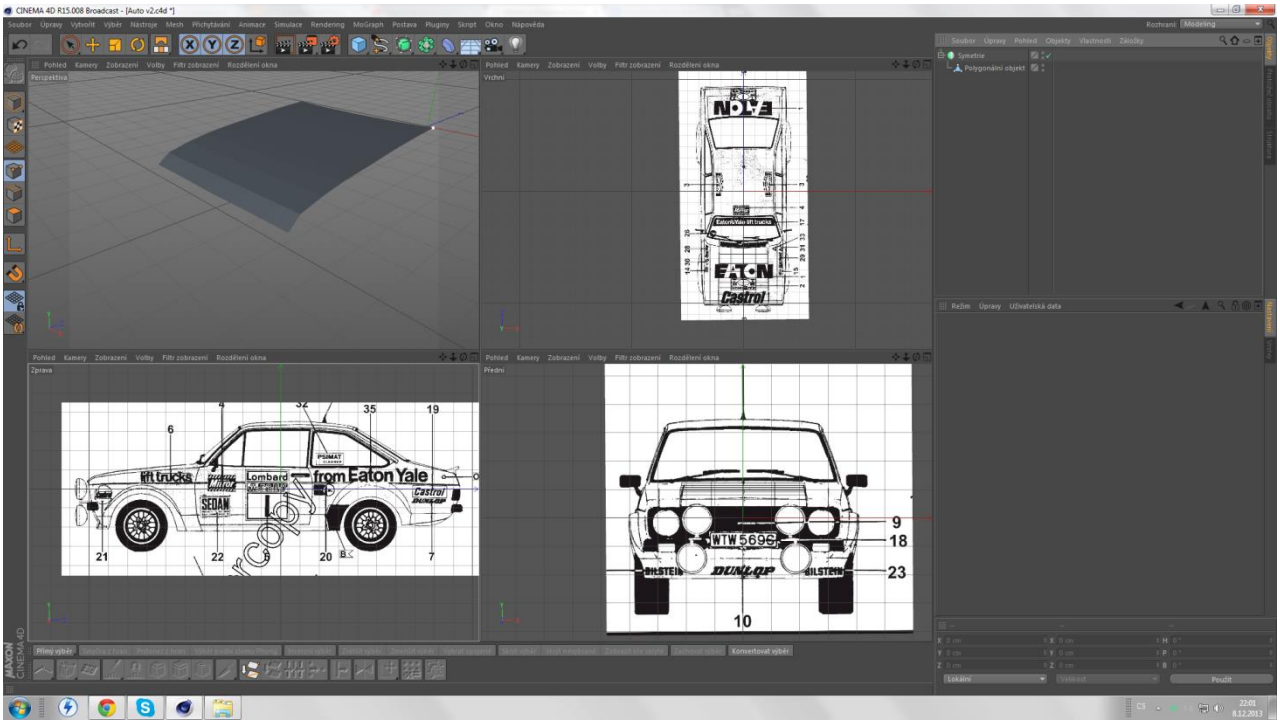
14 Popis Práce

Hlavní část práce byla realizována v programu Cinema 4D. Jeho studentskou verzi poskytla firma Maxon. Hlavní postup byl následovný: Modelování karoserie automobilu pomocí blueprintu a vymodelování kol. Vytvoření zjednodušeného modelu podvozku a odpružení podle reálné předlohy. Následoval proces zvaný rigging, ve kterém byla vozu dodána dynamika tzn. odpružení kol, motor, možnost otáčení kol a zatáčení.

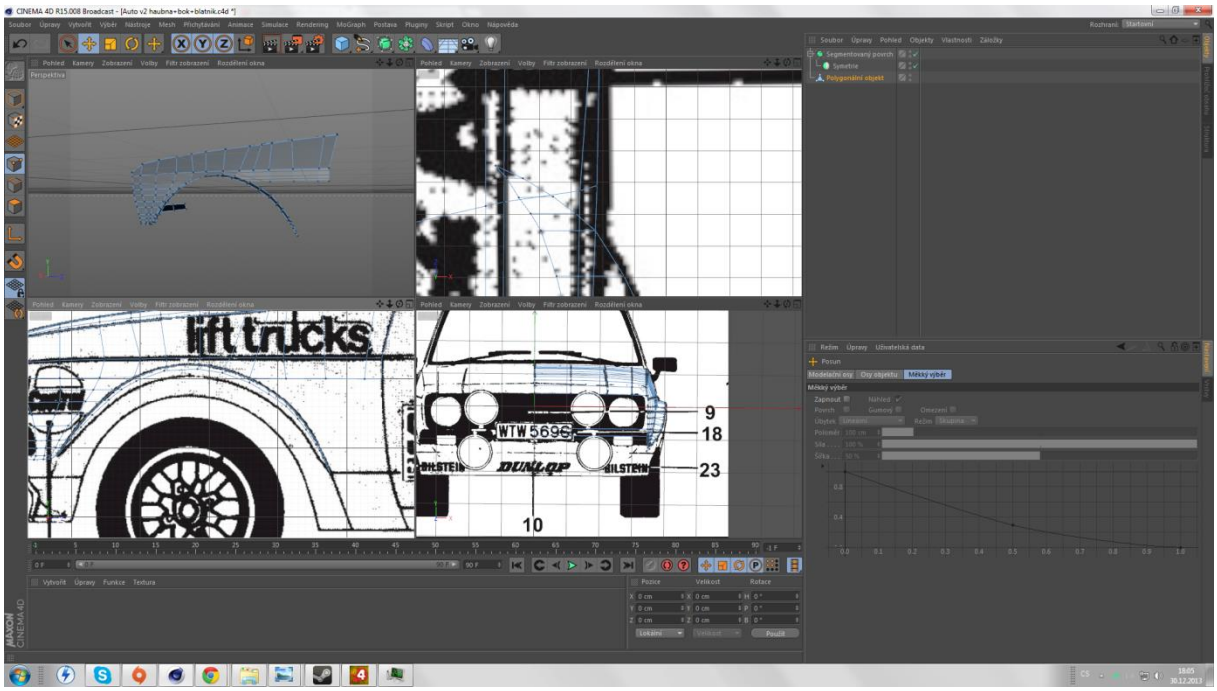
Takovýto model byl poté schopný pohybovat se po libovolném terénu. Pro tyto potřeby byla vymodelována testovací „dráha“, za použití motoru v autě a možnosti zatáčení se automobil rozpohyboval přes překážky a s implementací kamer byla výsledkem cca. dvou minutová animace.



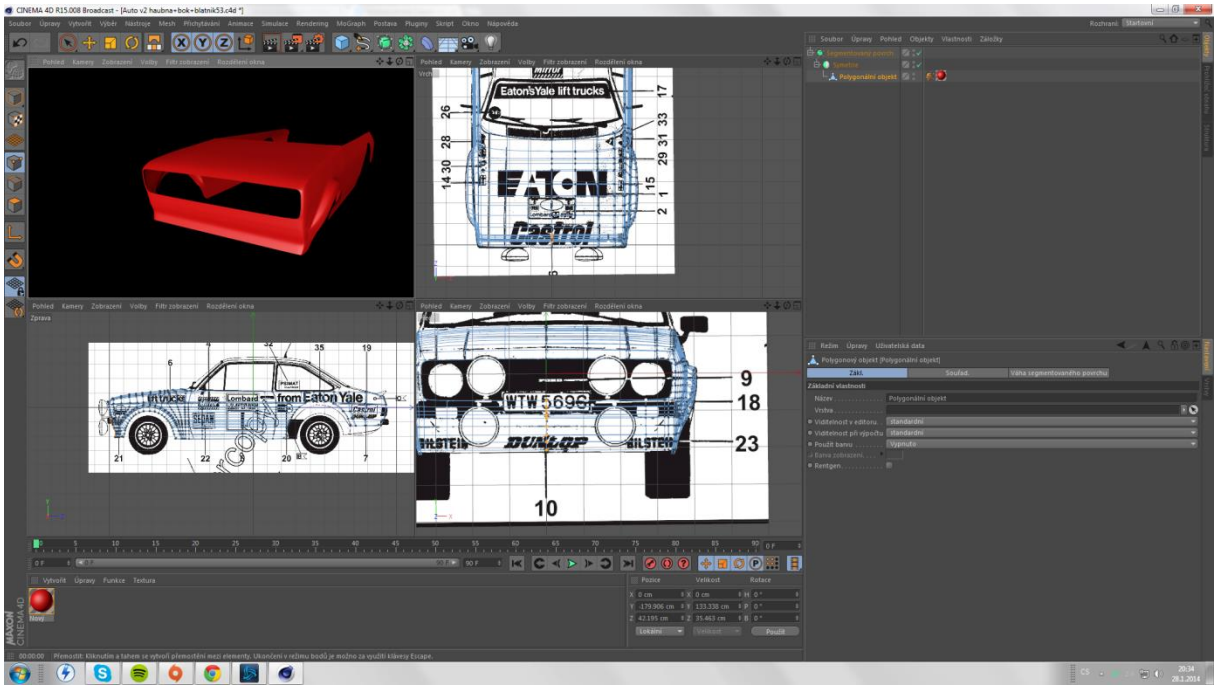
Začátek práce. Do programu Cinema 4D byly vloženy obrázky automobilu ze čtyř pohledů (tří, pohled zezadu se později vymění) – blueprinty.



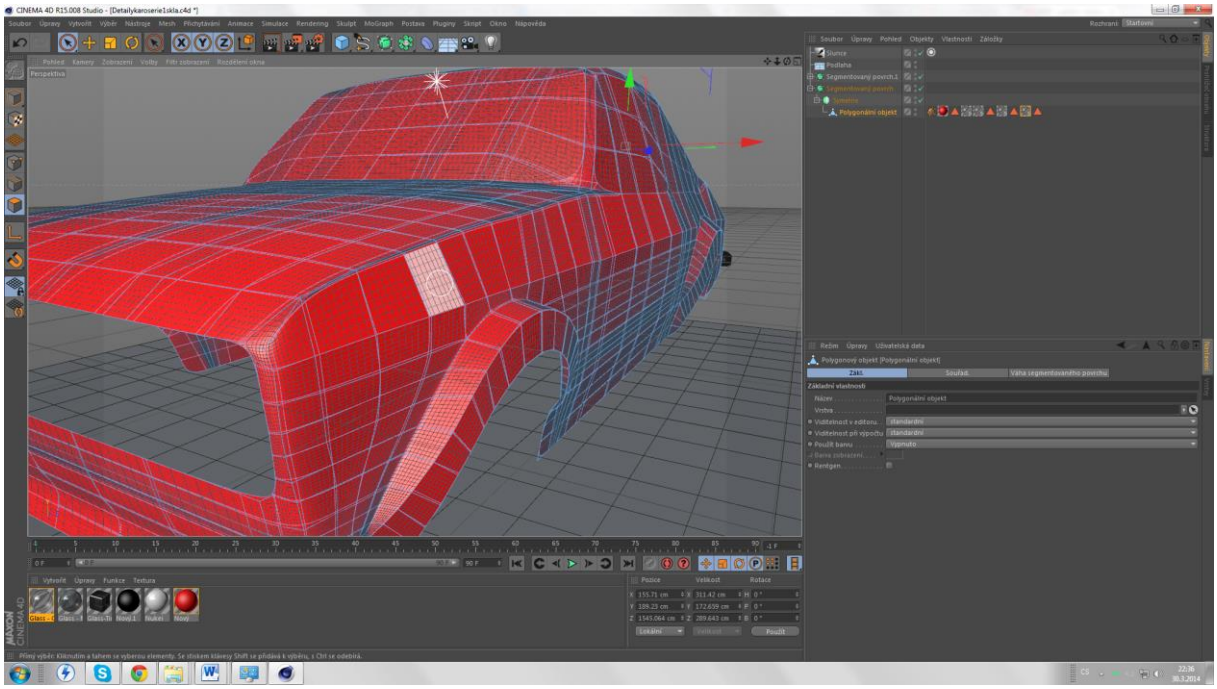
Pomocí polygonálního modelování započalo vytváření meshe modelu. Princip spočívá v tom, že je postupně přidáván bod za bodem, díky blueprintu je polohování bodů velice přesné, ale také velice zdlouhavé.



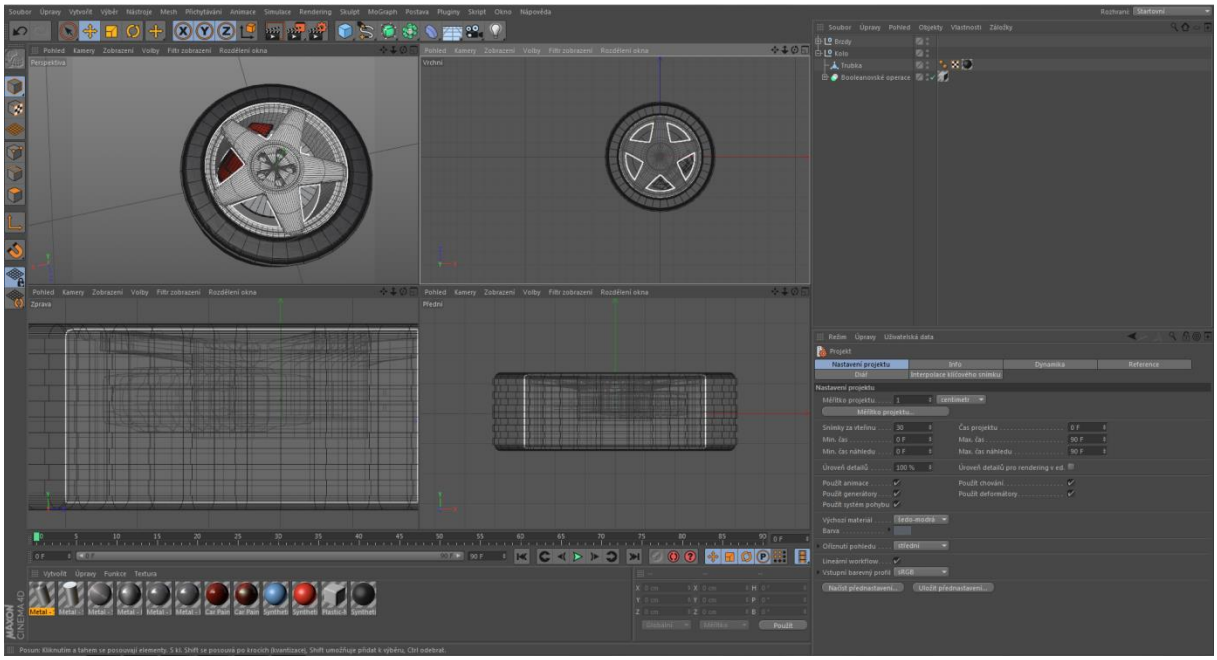
Na obrázku je možné vidět, jak mesh vypadá ze všech pohledů a jak překrývá blueprint. Detailnost modelu závisí na počtu polygonů v meshi tzn. čím více bodů, tím lépe bude model vypadat, zároveň ale bude jeho výpočet trvat déle. Proto rozhodnutí závisí na účelu práce, k animaci není potřeba tak detailní model jako třeba při modelování za účelem ultra realistického renderu.



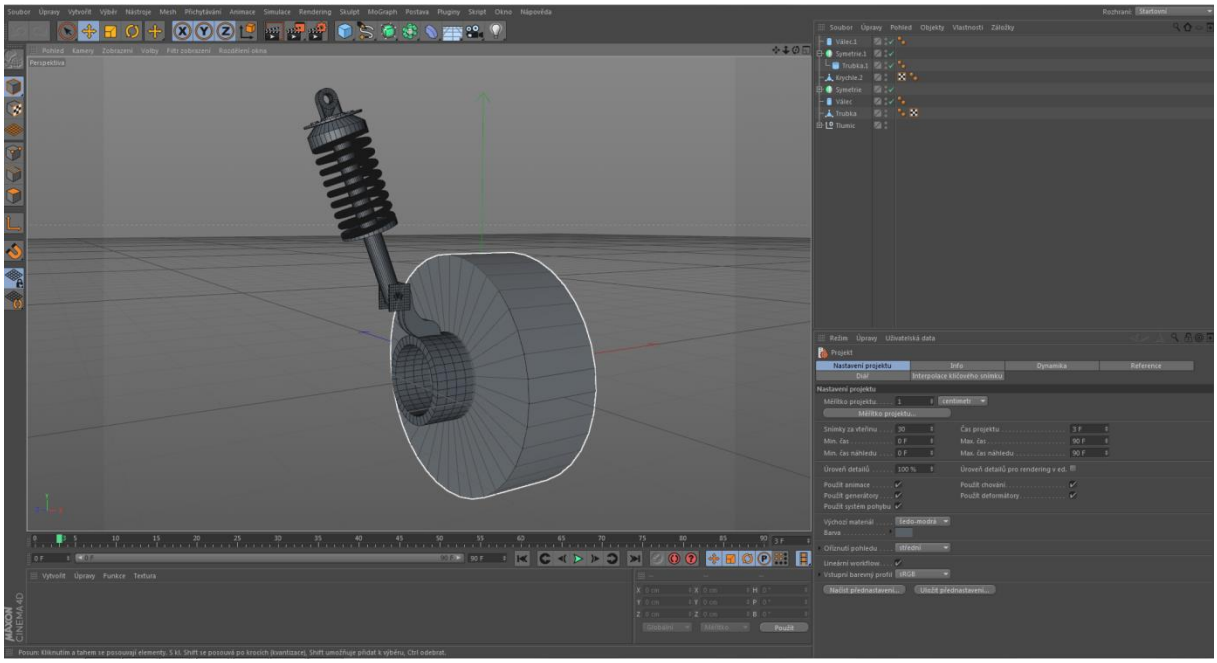
Pro větší detail modelu je možné využít funkci HYPERnurbs (Segmentovaný povrch), která počítačově zjemní (zaoblí) plochu modelu.



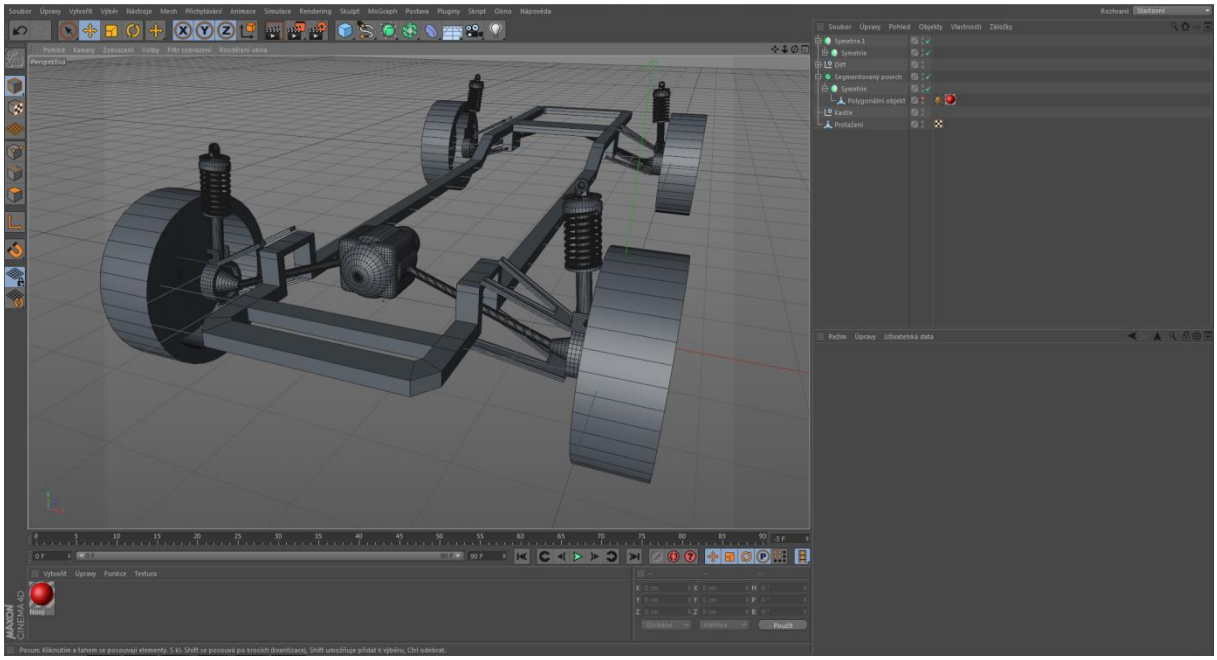
HYPERnurbs obsahuje i funkci, díky které je možné vybrané linky „zhranatět“, tudíž usnadňuje vytváření ostrých hran na jinak zaobleném povrchu modelu.



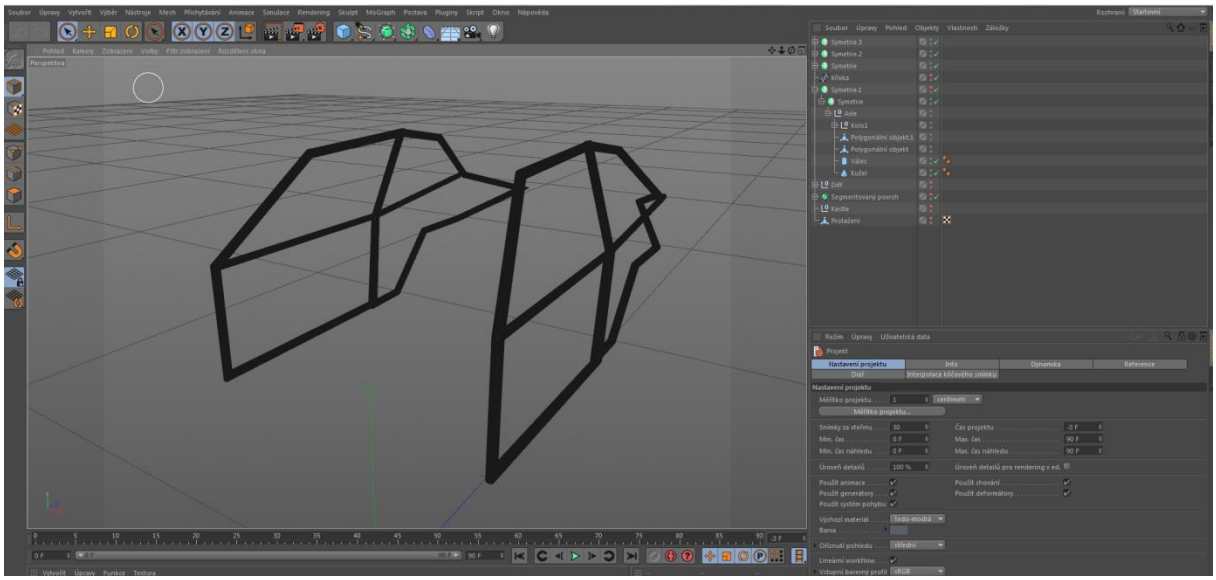
Po vytvoření karoserie automobilu bylo dalším krokem vytvoření disků kol, brzdových kotoučů, třmenů a pneumatik. Disky byly vytvořeny pomocí funkce „Pole“. Na modely byly rovnou naneseny materiály, které jsou buď vytvořené ručně či již obsažené v části Visualize.



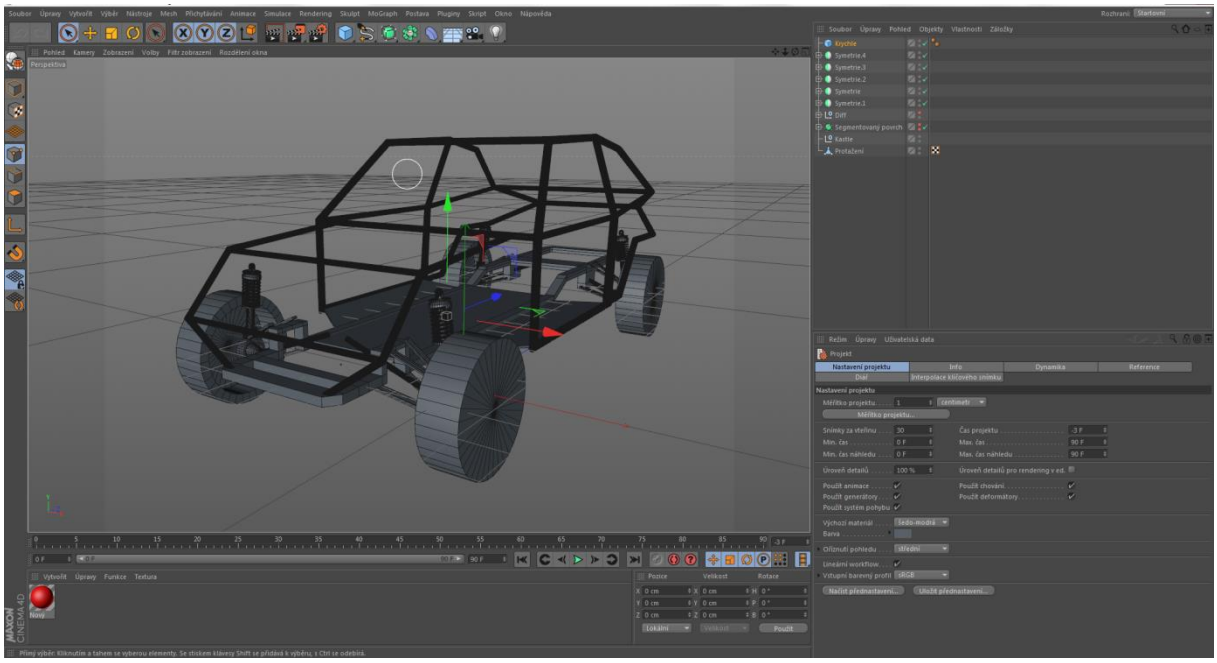
Dalším krokem bylo vymodelování podvozku viz. obrázek. Hlavní částí je tlumič, který hraje v konečné animaci velkou roli. K jeho vytvoření bylo klasicky použito funkce spirály a poté její protažení pomocí „protažení NURBS“.



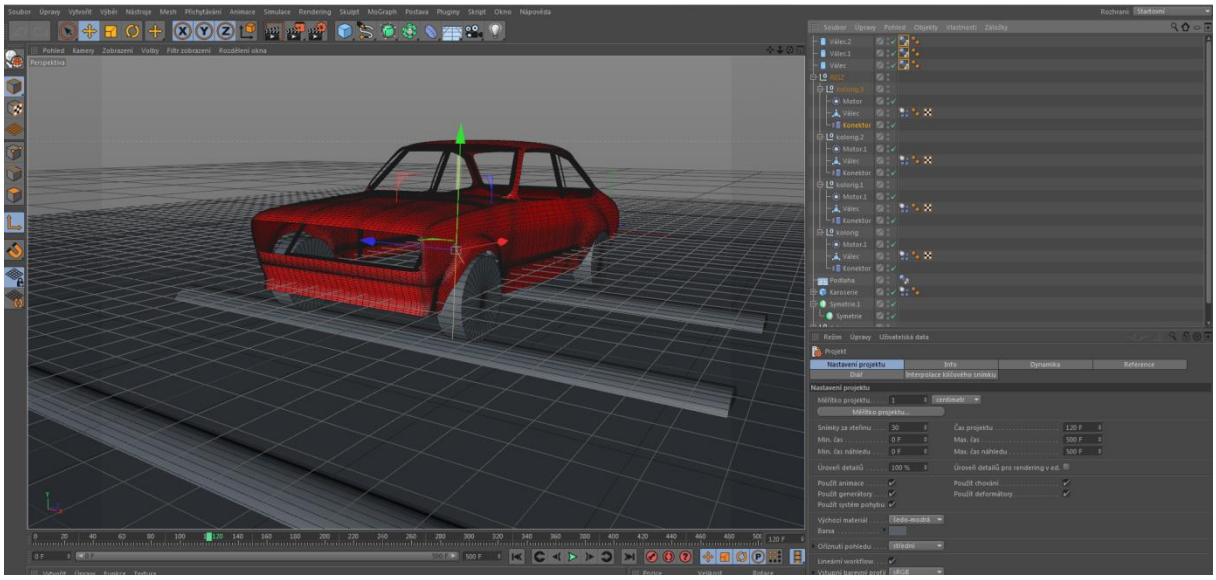
Při modelování podvozku se autor snažil, aby výsledek byl alespoň zjednodušeně podobný reálnému podvozku, který sice neobsahuje všechny součásti skutečného, ale ke znázornění pohybu bohatě postačuje.



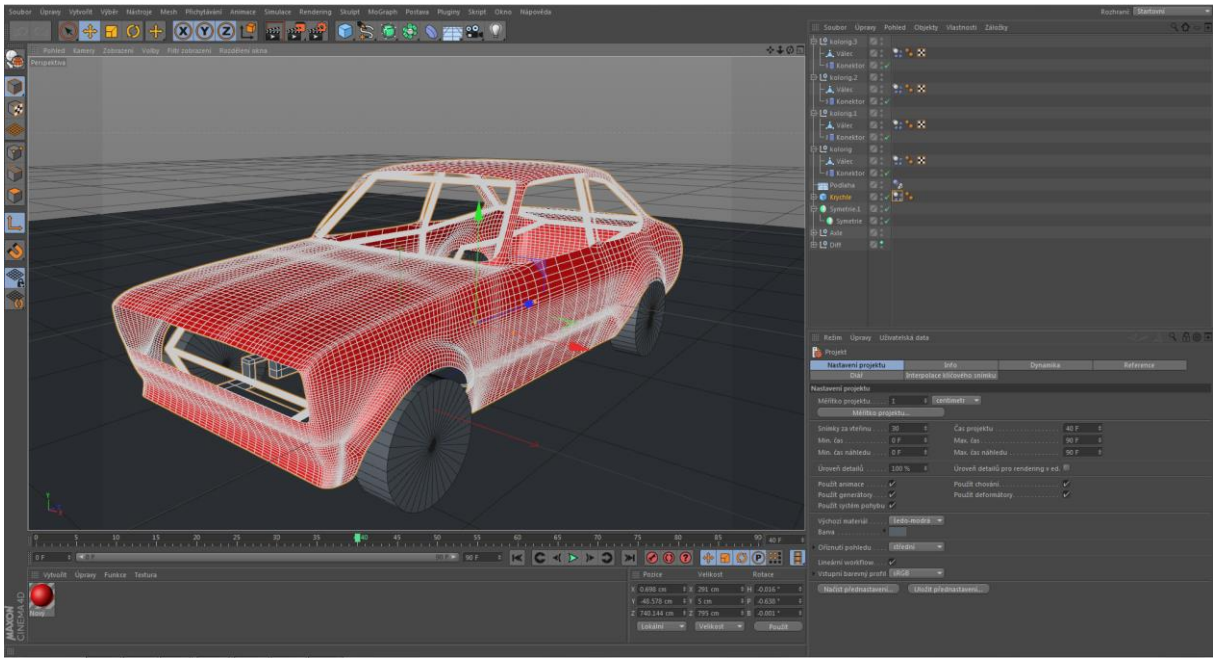
Tato klec byla vytvořena ze dvou důvodů. Jedním důvodem bylo, aby měly být tlumiče k čemu připevněné a druhým to, že předloha modelu je sportovní vůz používaný pro závody rallye, ve kterých se takovéto klece často používají. Technika použitá pro vytvoření tyčí, je klasické vytvoření spline (křivky), po které se poté protáhl kruh. To vše bylo podřízeno funkci symetrie.



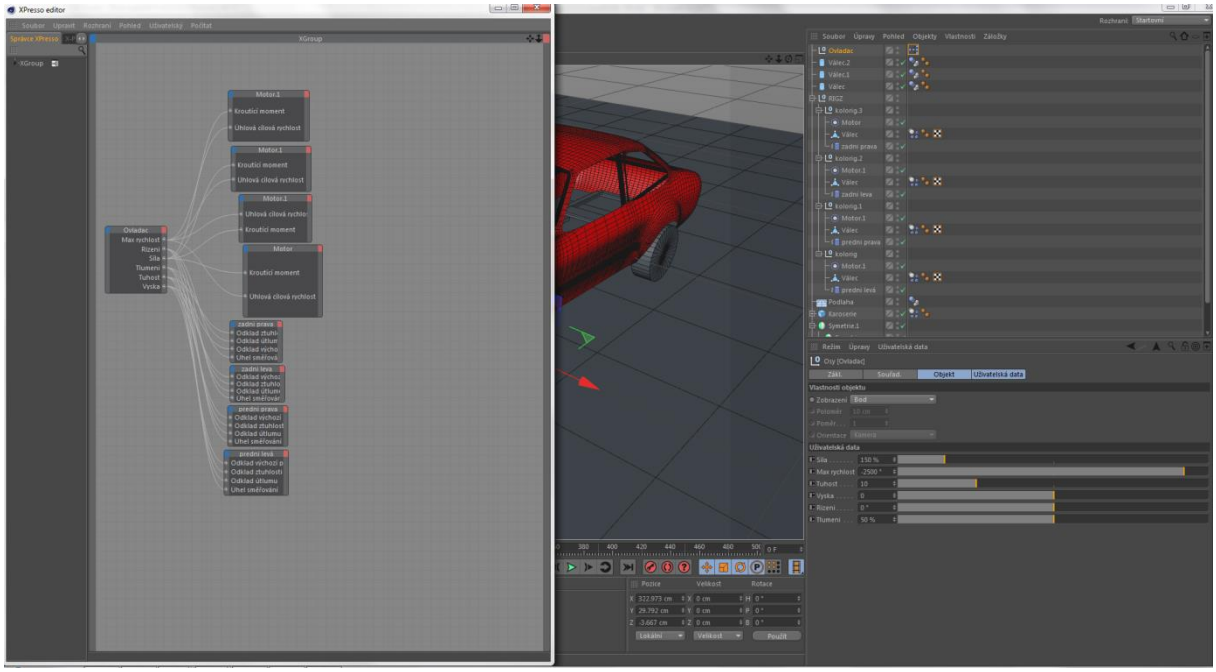
Takto vypadal model bez karoserie, když se spojily všechny součásti v jednom projektu. (Kola byly přidány až ke konci, kvůli rychlosti editoru).



Po dokončení modelu, bylo potřeba vytvořit „rig“ tudíž uchytit kola ke karoserii. To bylo provedeno pomocí funkce „Konektor“, kde byla poté vybrána funkce „Wheel Suspension“, která zajišťuje propojení všech čtyř kol s karoserií, na kterou byl nanesen „tag“ „Rigid body“(tuhé těleso).

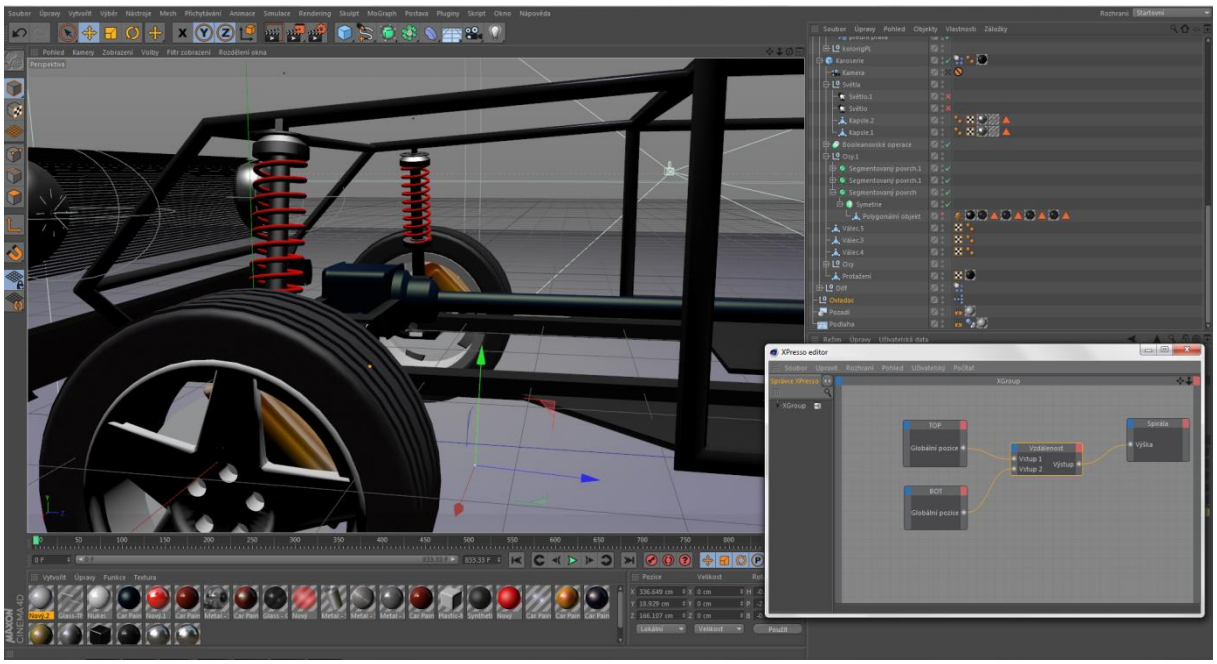


Pomocí funkce „motor“ bylo auto rozpohybováno. Bylo potřeba upravit nastavení otáček a síly motoru, ale i sílu odpružení, tlumení a ostatních parametrů zavěšení kol tak, aby se automobil choval co nejrealističtěji.

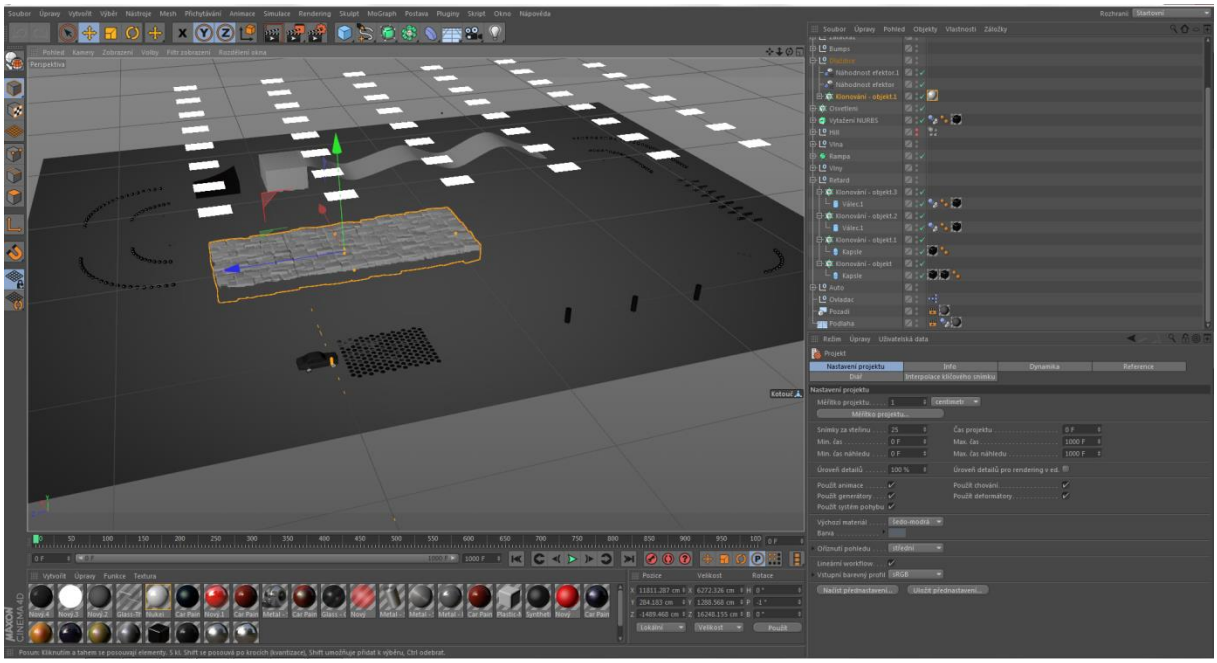


Pomocí modulu Xpresso byl vytvořen ovladač, který měl za úkol výrazně usnadnit nastavování parametru motoru, otáčení kol atd. Tudíž se dala jednoduše nastavit maximální rychlost, zrychlení, natočení kol, ale i výška pružin tlumiče a jeho tuhost.

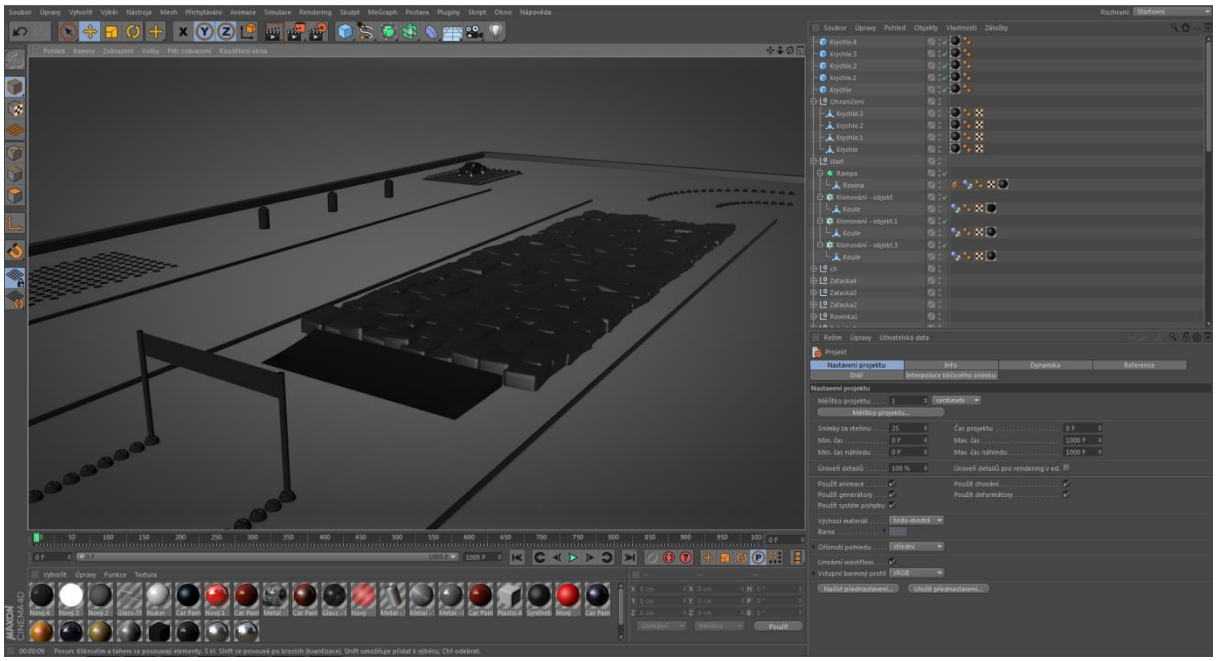
Po dokončení technické stránky věci, přešel autor ke stránce vizuální, tudíž byly na všechny povrchy naneseny materiály, buď vytvořené či přednastavené v části Visualize. Pro větší přesnost vytváření barev byla použita funkce „Slunce“, díky které byly materiály vidět v lepším světle.



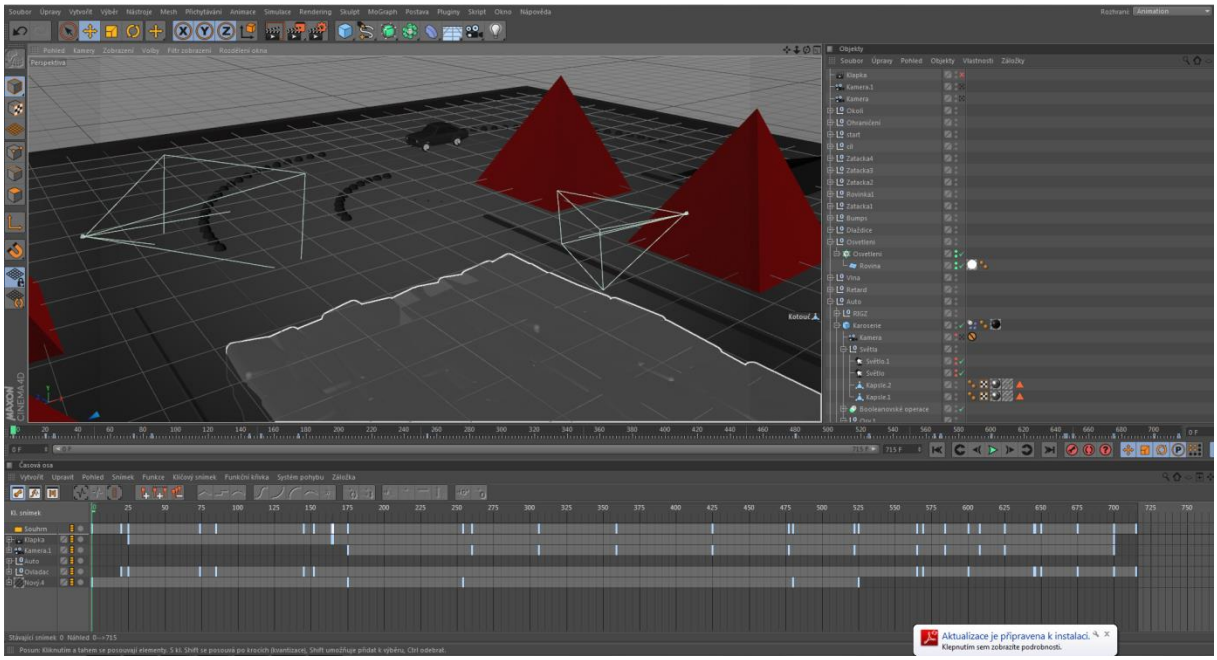
Jelikož dynamická část obsahovala pouze kola, musel autor „přidělat“ ostatní součásti podvozku tak aby se pohybovaly s nimi. Největším problémem byly pružiny v tlumičích, které se musely stlačit, když kolo najelo na nerovnost. K tomu bylo opět použito Xpresso, které dosti ulehčilo práci.



Když byl model automobilu plně funkční, začal autor pracovat na „překážkové dráze“. Bylo navrženo několik překážek tak, aby otestovaly funkčnost automobilu a nedělaly animaci jednotvárnou a nudnou, jak by tomu bylo u klasického přejíždění po pohoří.

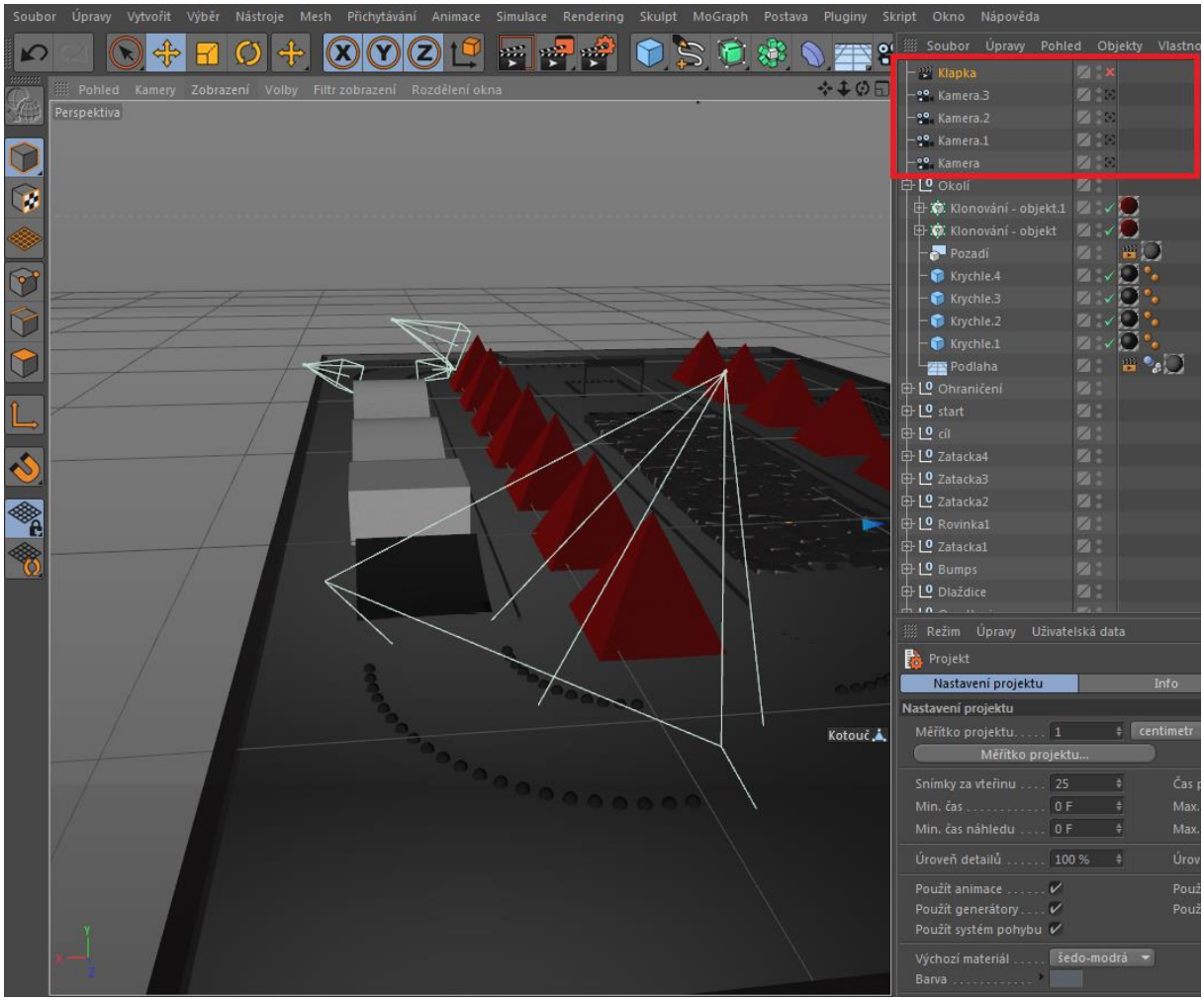


Celkově bylo prostředí navržené tak, aby bylo jednoduché (tzn. aby render netrval zbytečně dlouho). Z toho důvodu nebylo moc možností pro práci se světly. Největší překážka (viz. Obrázek) byla vytvořena pomocí funkce „pole“ a následného použití „efektoru náhodnosti“.

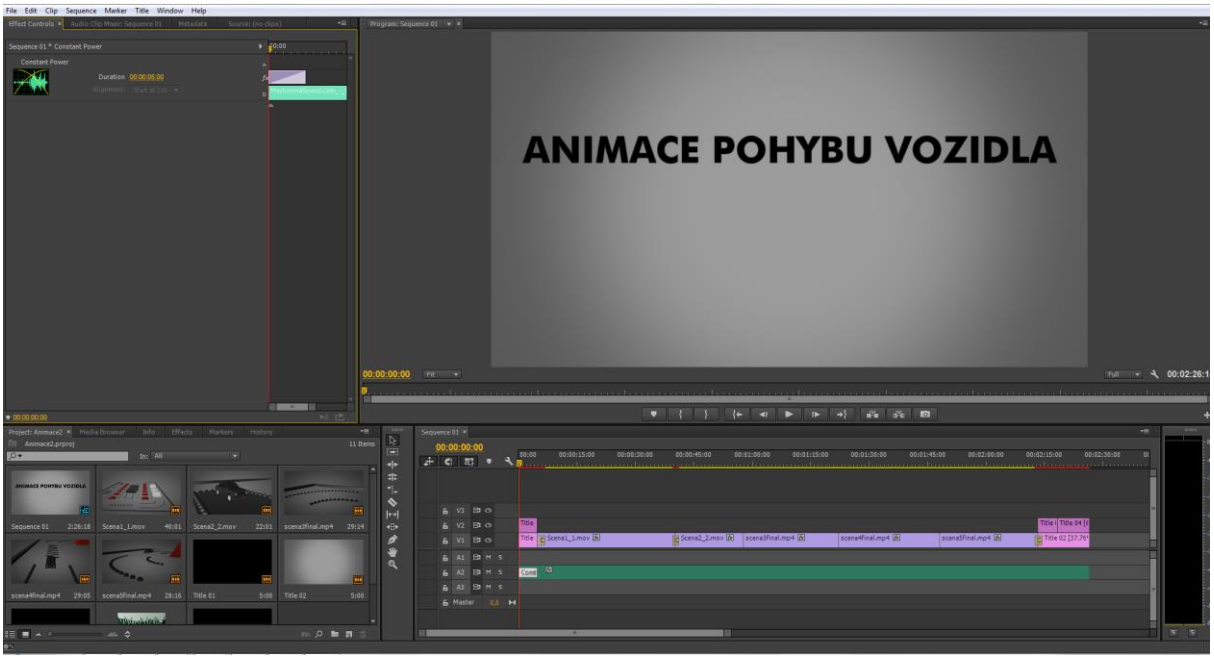


Po dokončení modelování prostředí, začal proces animace. Celá animace byla rozdělena do 5 scén, které se animovaly zvlášť a poté byly spojeny dohromady:

1. Scéna první. Představení prostředí a modelu automobilu.
2. Scéna druhá, kde automobil projíždí první a druhou překážku
3. Scéna třetí, kde automobil překoná třetí překážku s detailním záběrem na podvozek.
4. Ve čtvrté scéně, projede přes překážku čtvrtou na konci s menším skokem.
5. Na konec v páté scéně, projede vozidlo již zmiňovanou poslední překážkou a dojede do cíle



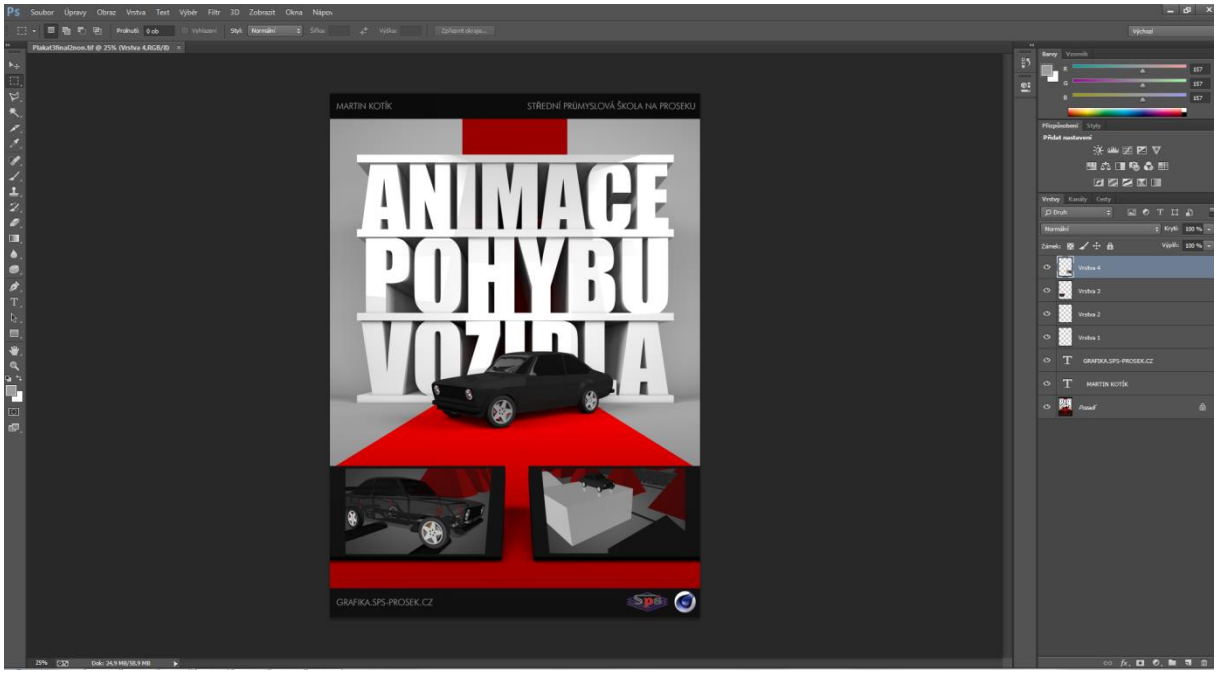
Pohyb kamer byl uskutečněn pomocí „keyframů“. Na rozdíl od klasického pohybu po křivce je pohyb pomocí klíčových snímků mnohem intuitivnější. Pohyb kamery totiž závisí přímo na pohybu uživatelovy obrazovky v čase. To znamená, že na prvním snímku uživatel zaznamená klíčový snímek, posune se v časové ose na libovolný snímek (např. třístý), posune kameru na libovolné místo a úhel pohledu a znovu zaznamená klíčový snímek. Po spuštění animace se kamera sama přesune z bodu A do bodu B v čase mezi klíčovými snímky.



Takto v podstatě probíhala animace celého vozidla. V čase se „keyframoval“ například úhel natočení kol, maximální rychlost atd.

Přepínání mezi kamerami v dané scéně bylo uskutečněno díky funkci „Klapka“, která umožňuje pomocí klíčových snímků měnit pohledy z kamer v čase, tudíž usnadňuje celkovou práci a není potřeba renderovat pohledy kamer v každé scéně zvlášť.

Po dokončení renderu všech pěti scén bylo potřeba sestříhat záběry tak, aby dohromady vytvořily plynulou animaci. To bylo uskutečněno pomocí zkušební verze Adobe Premiere Pro CC od společnosti Adobe. Scény byly „ořezány“ tak, aby nikde ve videu nebylo „hluché místo“. Na začátek a konec videa byly přidány titulní a závěrečná obrazovka. Přechody mezi některými scénami byly zjemněny pomocí „video transitions“ z knihovny programu. K videu byla také přidána zvuková stopa, která obohacuje celkový dojem z videa. Hudba je licencována pod licencí Creative Commons, která umožňuje její využití například ve videích jako je toto, pokud je autor hudby zmíněn.



Vytváření plakátu k maturitní práci probíhalo ve zkušební verzi programu Adobe Photoshop CC rovněž od společnosti Adobe. Základní pozadí bylo navrženo a vyrenderováno v programu Cinema 4D, z něj byly rovněž použity obrázky nacházející se v plakátu. Obrázky animace a loga byly oříznuty a umístěny na místo.

15 Seznam použitých zdrojů

- JIŘÍ, Žára a Jiří ŽÁRA. *Moderní počítačová grafika*. Vyd 1. Brno: Computer Press, 2004, 609 s. ISBN 80-251-0454-0.
- *3D grafika* [online]. 2014 [cit. 2014-03-31]. Dostupné z: http://www.3dgrafika.cz/
- *Grafika.cz: vše o počítačové grafice* [online]. 2014 [cit. 2014-03-31]. Dostupné z: http://www.grafika.cz/
- *About.com* [online]. 2014 [cit. 2014-03-31]. Dostupné z: http://3d.about.com
- Wikipedie: Cinema 4D. [online]. [cit. 2014-03-31]. Dostupné z: http://cs.wikipedia.org/wiki/Cinema_4D
- Wikipedie: Computer aided design. [online]. [cit. 2014-03-31]. Dostupné z: http://cs.wikipedia.org/wiki/Computer_aided_design
- Wikipedie: Počítačová 3D grafika. [online]. [cit. 2014-03-31]. Dostupné z: http://cs.wikipedia.org/wiki/3D_grafika
- Wikipedia: Rendering. [online]. [cit. 2014-03-31]. Dostupné z: http://en.wikipedia.org/wiki/Rendering_(computer_graphics)
- Wikipedia: Cartesian coordinate system. [online]. [cit. 2014-03-31]. Dostupné z: http://en.wikipedia.org/wiki/Cartesian_coordinate_system
- *Adobe* [online]. 2014 [cit. 2014-03-31]. Dostupné z: http://www.adobe.com/
- *3D Guru* [online]. 2014 [cit. 2014-03-31]. Dostupné z: http://www.guru3d.com/
- Wikipedia: Computer animation. [online]. [cit. 2014-03-31]. Dostupné z: http://en.wikipedia.org/wiki/Computer_animation
- Wikipedia: OpenGL. [online]. [cit. 2014-03-31]. Dostupné z: http://cs.wikipedia.org/wiki/OpenGL
- Wikipedia: OpenCL. [online]. [cit. 2014-03-31]. Dostupné z: http://cs.wikipedia.org/wiki/OpenCL
- Wikipedia: CUDA. [online]. [cit. 2014-03-31]. Dostupné z: http://cs.wikipedia.org/wiki/CUDA
- *The Blueprints* [online]. 2014 [cit. 2014-03-31]. Dostupné z: http://www.the-blueprints.com/
- *Maxon: 3D for the real world* [online]. 2014 [cit. 2014-03-31]. Dostupné z: http://www.maxon.net/