



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

VIRTUÁLNÍ KOVÁŘ

VIRTUAL BLACKSMITH

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. MÁRIA HALAMOVÁ

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. TOMÁŠ MILET, Ph.D.

BRNO 2022

Zadání diplomové práce



Studentka: **Halamová Mária, Bc.**
Program: Informační technologie
Obor: Počítačová grafika a multimédia
Název: **Virtuální kovář**
Virtual Blacksmith
Kategorie: Počítačová grafika
Zadání:

1. Seznamte se s herním enginem Unity a virtuální realitou. Nastudujte techniky deformace a vizualizace vhodné pro modelování kujných materiálů.
2. Navrhněte virtuální výukovou aplikaci umožňující demonstraci kovářských technik.
3. Implementujte navrženou aplikaci.
4. Zhodnoťte výsledky, vytvořte demonstrační video, navrhněte rozšíření.

Literatura:

- Kevin Mack, Robert Ruud.: Unreal Engine 4 Virtual Reality Projects: Build immersive, real-world VR applications using UE4, C++, and Unreal Blueprints. Packt Publishing Ltd, 2019. ISBN-13: 978-1789132878.

Podrobné závazné pokyny pro vypracování práce viz <https://www.fit.vut.cz/study/theses/>

Vedoucí práce: **Milet Tomáš, Ing., Ph.D.**
Vedoucí ústavu: Černocký Jan, doc. Dr. Ing.
Datum zadání: 1. listopadu 2021
Datum odevzdání: 18. května 2022
Datum schválení: 1. listopadu 2021

Abstrakt

Táto práca sa zaoberá tvorbou aplikácie na demonštráciu manuálneho kováčstva pre platformy virtuálnej reality. Táto aplikácia má za účel predstaviť historické kováčstvo pomocou moderných technológií z pohľadu prvej osoby. Práca skúma spôsoby realistickej simulácie kováčskych techník, ktoré sa v tomto remesle používajú a vhodnú metódu pre vizualizáciu kovového objektu. Aplikácia implementuje 3D simuláciu nahrievania a deformácie kujných materiálov za pomoci kováčskych nástrojov ovládaných užívateľom vo virtuálnej realite. Parciálne nahrievanie kovu je vizualizované pomocou procedurálneho materiálu. Užívateľský zážitok rozširuje pridaná zvuková a haptická odozva. Výsledná aplikácia je vytvorená vo frameworku Unity a je určená pre set virtuálnej reality nazývaný Oculus Rift.

Abstract

This thesis deals with creation of an application for demonstration of smithing for virtual reality platforms. The purpose of the application is to present historical smithing craft with the help of modern technologies from the first person's view. The thesis researches realistic simulation methods used in smithing techniques and an appropriate way of visualising a metal object. The application implements a 3D simulation of heating and deformation of forging materials using user-controlled forging tools in virtual reality. Partial heating of the metal is visualized by procedural material. The user experience is enhanced by added audio and haptic feedback. The final application is created in Unity framework and is developed for Oculus Rift virtual reality set.

Klíčové slová

Unity, virtuálna realita, Blender, kováčsky simulátor, deformácia meshu, Oculus Rift

Keywords

Unity, virtual reality, Blender, blacksmith simulator, mesh deformation, Oculus Rift

Citácia

HALAMOVÁ, Mária. *Virtuální kovář*. Brno, 2022. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Tomáš Milet, Ph.D.

Virtuální kovář

Prehlásenie

Prehlasujem, že som túto diplomovú prácu vypracovala samostatne pod vedením pána doktora Tomáša Mileta. Uviedla som všetky literárne pramene, publikácie a ďalšie zdroje, z ktorých som čerpala.

.....

Mária Halamová

25. mája 2022

Podakovanie

Chcela by som poďakovať vedúcemu práce doktorovi Tomášovi Miletovi za vedenie práce, rady a pravidelné konzultácie. Taktiež by som chcela poďakovať rodine, ktorá ma vždy podporuje, kamarátom za testovanie aplikácie a MCR za psychickú podporu.

Obsah

1	Úvod	2
2	Prehľad	4
3	Podobné kovácke aplikácie	7
4	Kováčstvo	14
4.1	Základy kováčstva	14
4.2	Nástroje	14
4.3	Techniky	18
5	Technológie	22
5.1	Unity 3D	22
5.2	Virtuálna realita	31
5.3	Blender	34
6	Návrh aplikácie	36
6.1	Koncept	36
6.2	Kováčstvo	39
6.3	Scéna, Modely, Materiály	43
6.4	Interakcia a navigácia	49
6.5	Deformácia materiálu	51
6.6	Doplňkové efekty	55
7	Implementácia	56
7.1	Tvorba 3D objektov	58
7.2	Unity Setup	59
7.3	Herné objekty	60
7.4	Behaviorálna časť aplikácie	69
8	Záver	76
	Literatúra	79
9	Obsah priloženej SD karty	81

Kapitola 1

Úvod

V minulosti boli kováčstvo (obr. 1.1), drevárstvo (stolárstvo) a mnohé iné remeslá veľmi rozšírené a ich výrobky ľudia používali v každodennom živote. Keď potrebovali rôzne predmety, ľudia nemali možnosť zájsť si do veľkoobchodov alebo špecializovaných obchodov a nakúpiť tam. Museli si buď veci každodennej potreby sami vyrobiť, alebo objednať ich výrobu u špecializovaných remeselníkov. Od výroby predmetov dennej potreby ako sú malé príbory, cez pracovné nástroje ako lopaty, motyky a sekery, až po veľké komplexné nástroje alebo zbrane. Špecializovaní remeselníci boli veľmi dôležití a vyhľadávaní. V súčasnosti takéto remeslá ustupujú do úzadia. Vo väčšine výrobných procesov nahradili ľudskú prácu automatické mechanické stroje a remeselnícka práca sa považuje za historické dedičstvo. Tieto remeslá sa uchovávajú najmä vďaka malému množstvu jedincov, ktorí buď prevzali dané remeslo v rodine, alebo majú oň silný záujem, nekonvenčný v dnešnej dobe.



Obr. 1.1: Historická kováčska dielňa s tradičnými kováčskymi nástrojmi¹

¹<https://www.oldfieldforge.co.uk/history-of-blacksmithing>

Mládež často ani nevie, že takéto remeslá existovali a ako ťažké bolo ručne vytvoriť predmety, ktoré sú dnes široko dostupné. Nevedia aké množstvo rôznych faktorov je potrebné brať do úvahy, aby dosiahli želaný výsledok. Niektorí sa ku výrobe nástrojov (predovšetkým zbraní) dostanú pri hrách, ktoré ale majú malú výukovú hodnotu. V týchto hrách typicky stačí stlačiť jedno tlačítko a z kúska železnej rudy sa stane meč. Mládeži je však možné ukázať zábavnou formou prostredie kde sa remeslo vykonávalo, nástroje a samotné techniky používané pri kutí kusu železa do požadovaného tvaru výsledného predmetu.

Táto aplikácia, ako výsledok tejto diplomovej práce, demonštruje činnosť kováčov pri deformovaní kusu železného kovu na výrobu mečov z historického pohľadu. Možnosť využitia v budúcnosti je na prezentáciu remesla a niektorých fyzikálnych javov či už v školách alebo historických expozíciách na hradoch a zámkoch, prípadne v hrách. Je čoraz ťažšie zaujať deti históriou a prehliadky na hradoch a zámkoch už neudivujú svojou nevšednosťou, keď podobný hrad môžu nájsť v počítačových hrách, dokonca si ho sami vybudujú podľa svojich predstáv. Je taktiež ťažšie udržať ich pozornosť pri vyučovaní a využívanie interaktívnej tabule a rôznych aplikácií sú v školách čím ďalej tým viac rozšírené. Táto aplikácia vo virtuálnej realite poskytuje väčší potenciál ako zaujať dnešné deti a mládež a môže pomôcť aj atraktivnosti expozícií hradov, zámkov, múzeí či skanzenov. Ako už bolo spomenuté, školy sa musia snažiť čoraz viac aby vzbudili a udržali záujem deti a táto aplikácia je v škole využiteľná najmä ako interaktívna časť učiva dejepisu alebo pracovnej výchovy, poprípade vysvetlenia niektorých fyzikálnych pojmov.

Spojením kováčstva a virtuálnej reality (VR) môže vzniknúť unikátna a zaujímavá aplikácia, ktorá poskytuje dosť rôznych možností využitia z hľadiska rôznych kováčskych techník, poprípade ich častí. Vývojár má teda pomerne veľkú voľnosť pri návrhu a môže si vybrať, v akom pohľade predstaví v aplikácii kováčstvo pre užívateľa.

Je možné sa zamerať na vytvorenie takej aplikácie, ktorá vyobrazuje postup jednotlivých častí viac všeobecne, alebo detailnejšie spracovať niektorú konkrétnu časť. Ako príklad môže byť zameranie sa na realistickejšie deformovanie kovaného materiálu a dať užívateľovi voľnosť v tom, čo a ako môže v aplikácii vytvoriť.

Takýto software skombinovaný s technológiou ako je virtuálna realita môže predstavovať viac ako aplikáciu určenú pre zábavu. Bežne dostupné VR sety umožňujú čiastočné alebo plné ponorenie, čím výrazne zosilnia realistický zážitok. V takejto aplikácii možno potom naprogramovať realistický systém kovania, ktorý bude užívateľ ovládať cez ručné ovládače, s ktorými bude schopný držať virtuálny kus železnej tehly a kováčskeho kladiva. Takýto systém má potom schopnosť mať určené správanie pri strete kovu s kladivom, a práve užívateľ bude plne ovplyvňovať silu jeho úderu, nasmerovanie a polohu. Vďaka tomu môže aplikácia splniť nielen zábavný účel, ale aj výukový a prípadne experimentálny.

Kapitola 2

Prehľad

Ako už bolo spomenuté, táto diplomová práca je zameraná na návrh a tvorbu netradičnej aplikácie za účelom demonštrovania historického kováčstva. Za pomoci moderných technológií a čoraz populárnejšej virtuálnej reality je možné takýto zážitok preniesť záujemcom priamo do ich rúk.

Virtuálny kováč teda má za snahu zobrazit' čo najrealistickejší virtuálny model deformovania kovových materiálov. Zložité deformovanie objektov v reálnom čase na základe presných fyzikálnych modelov je stále veľmi obtiažne z dôvodu výpočtových a pamäťových limitov dostupných VR zariadení, preto je aj pre tento dôvod zohľadnený spôsob implementácie. Vďaka tomu môže aplikácia fungovať dobre nielen na high-end VR setoch.



Obr. 2.1: Ukážka výslednej scény aplikácie

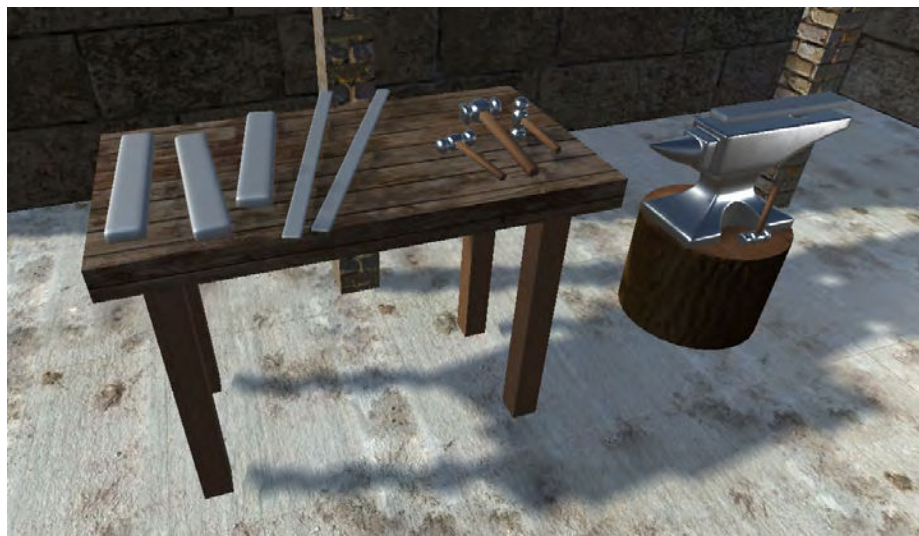
Scéna aplikácie je navrhnutá tak, aby vyhovovala väčšiemu okruhu užívateľov z pohľadu VR ovládania. Je menšia a tak nie je potrebná veľká miestnosť a priestor, ani priveľmi využívať pohyb a otáčanie pomocou kontrolérov (čo môže u niektorých užívateľov spôsobiť závrate). Virtuálne prostredie tejto kováčskej dielne obsahuje všetky potrebné základné objekty pre demonštráciu kovania. Medzi ne patria pec, nákova, kladivá, kovové objekty určené na deformáciu a ďalšie predmety (obr. 2.1).

Aplikácia umožňuje nahrievať kovové objekty v peci. Tento jav je v aplikácii vizualizovaný na základe gradientu reprezentujúceho teploty kovu. Časti kovu, ktoré sú vložené do ohňa, sú s rastúcim časom zahrievané na kujnú teplotu. V strede ohňa sa kov nahrieva rýchlejšie než na okrajoch. Tá časť kovu, ktorá v ohni nie je sa pomaly chladí ale nie tak rýchlo ako keď je celý kovový objekt mimo ohňa. Keďže prirodzené ochladzovanie kovu trvá istú dobu, pre rýchle schladenie po dokončení kovania je možné použiť nádobu s vodou (obr. 2.2).



Obr. 2.2: Pec s ohňom vytvorený ako časticový efekt na nahrievanie kovu a nádoba s vodou na schladenie kovu

Nahriate kovové objekty je možné deformovať pomocou kladív ak sú držané alebo položené na nákove (obr. 2.3). Kovový objekt sa deformuje na základe toho kde je udierané kladivom. Deformácia sa aplikuje na celý objekt aby sa zachovával objem. Ak sa kov deformuje dovnútra objektu pri údere kladivom, rovnaký objem sa rozloží do zväčšenia celého kovového telesa mimo oblasti úderu kladiva. Užívateľ teda nie je obmedzený vopred pripraveným tvarom zbrane ku ktorej by sa pomaly približoval pomocou deformácie. Je teda teoreticky možné sa dostať do ľubovoľného stavu na základe užívaľovej zručnosti. Sila deformácie je ovplyvnená teplotou danej časti kovového objektu a vzdialenosťou tejto časti od miesta kolízie s kladivom.



Obr. 2.3: Vľavo stôl s kovovými objektami a kladivami na ich deformáciu, vpravo náková na ktorej sa kovy deformujú

Pre dotvorenie celkového obrazu sú použité audiovizuálne zložky. Do vizuálnej časti patrí oheň a voda. Oheň je vytvorený pomocou častíc a voda pomocou procedurálneho materiálu. Do zvukovej časti patrí napríklad zvuk ohňa alebo zvuky udierania kladiva o kov.



Obr. 2.4: Stôl na ukladanie hotových výrobkov

Hotové výrobky je možné vystaviť na odkladacom stole. Nepotrebné kovy poprípade nevydarené výrobky je možné zahodiť do koša a tým ich zmazať (obr. 2.4). Nové kovy sa generujú na stole po tom ako si ich užívateľ odtiaľ vezme.

Kapitola 3

Podobné kováčske aplikácie

V tejto kapitole sú predstavené VR aplikácie, ktoré zahŕňajú kováčstvo ako časť svojho obsahu. Popísané sú z pohľadu kováčstva, čo korešponduje s témou tejto práce. Popis je teda zameraný najmä na možnosti deformácie kovov.

Hammer & Anvil VR

Hammer & Anvil je VR kováčsky simulátor. Táto hra umožňuje kovať sekery, dýky, oštep, štíty a meče zo železných palíc a iných základných materiálov. V hre je možné používať pec, nákovu, brúsny kameň a kladivo na výrobu zbraní. Vybavenie je možné zlepšovať pomocou herných peňazí. Cieľom hry je dokončiť objednávky zákazníkov v časovom limite a tým si zarábať peniaze¹.

Hra je určená pre VR sety Valve Index a HTC Vive, na iných zariadeniach (napríklad od značky Oculus) hra nefunguje. Herná scéna je malej veľkosti, a tak sa užívateľ nemusí veľmi pohybovať, stačí sa iba otáčať.

Proces deformácie začína kusom kovu tvaru kocky, ktorý sa vloží nahriať do pece (obr. 3.1a). V peci po zahriatí zmení farbu (obr. 3.1b). Po vytiahnutí z pece a priložení ku nákovu je možné doň udrieť približne 5x pred tým než sa znova schladí a stane sa nedeformovateľný. S každým úderom sa mení tvar kovového predmetu a stáva sa ploskejší zo strany úderu kladiva. Môže sa rozširovať v zvyšných dvoch smeroch (obr. 3.1c, 3.1d). Na sile úderu nezáleží, kov sa posúva postupne po stavoch ktoré sú predpripravené. Takto deformovaný kus kovu dosiahne tvar kvádra. Aby sa z kvádra stal meč, sekera, dýka alebo iný nástroj, priloží sa kovový kváder ku brúsnemu kameňu (obr. 3.1e) a po chvíli sa zmení na objekt predpripraveného tvaru (obr. 3.1f, 3.1g). Teda ak boli údery iba z jednej strany 5x, po priložení ku brúsnemu kameňu sa premení kváder kovu na hlavu sekery. Podobne aj na ostatné typy objektov ponúkaných v hernom obchode. Na dotvorenie stačí priložiť kváder predstavujúci drevo a tieto 2 objekty sa zmenia na celú sekeru alebo iný predmet (obr. 3.1h).

Township Tale - Blacksmithing

V tejto hre sa spracovávajú surové materiály na kovové bloky a tie na odliate hrubé násady na nástroje a zbrane vo veľkej peci (obr. 3.2a). Stačí vložiť materiál do pece popripade použiť šablónu a zvýšením teploty v peci vypadnú hotové výrobky z opačnej strany. Kovové

¹https://store.steampowered.com/app/1017780/Hammer__Anvil_VR/

²<https://youtu.be/yulMcIQVKyE>, https://store.steampowered.com/app/1017780/Hammer__Anvil_VR/



(a) Nahrievanie v peci



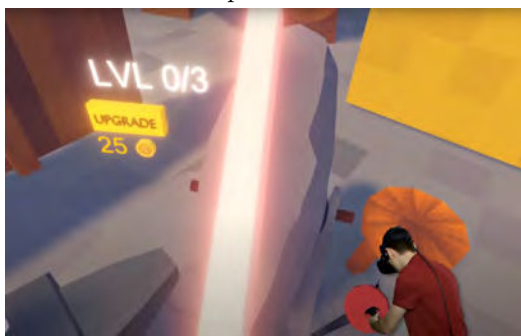
(b) Nahriata kocka kovu držaná v ruke pri nákov



(c) Kocka kovu po 1. údere sa zmenila na kváder a stratila teplotu v čase



(d) Deformácia kovu po 4. údere, kváder je plochý a už je potreba kov znova nahriať



(e) Priloženie kovu ku brúsne kameňu



(f) Náhla premena kovu na meč



(g) Výsledný vyrobený meč



(h) Ukážka hotovej sekery odovzdanej zákazníkovi a vľavo základná kocka kovu s ktorou sa začína

Obr. 3.1: Ukážka z hry Hammer & Anvil²

odliate časti sú hrubé a nemajú ostrie (obr. 3.2b), takže ich treba nahriať v ohni/pahrebe (obr. 3.2c). Potom ich treba vziať ku nákovu a po pár úderoch kladivom sa zmenia do finálnej ostrej podoby (obr. 3.2d, 3.2e).

Craft Keep VR

V hre Craft Keep VR sa užívateľ stará o vyrábanie zbraní, nástrojov a elixírov. Na výrobu zbraní a nástrojov sa najskôr roztavia rudy v peci do tekutého stavu (obr. 3.3a). Tekutý kov sa potom vyleje do šablóny (obr. 3.3b) a po udretí kladivom vznikne výsledný produkt (obr. 3.3c). Ak však užívateľ neudrie správnu silou, môže sa stať že výrobok bude mať v sebe diery (obr. 3.3d). V tomto prípade sa výrobok vloží do pece (obr. 3.3e), ktorá ho nahreje a už vtedy opticky vyplní diery. Tento nahriaty produkt už stačí iba položiť na nákovu, niekoľkokrát udrieť kladivom (obr. 3.3f) a nakoniec spojiť s násadou.

Fantasy Smith VR

Fantasy Smith VR je hra ktorá umožňuje užívateľovi aspoň trochu voľnosti vo vyrábaní zbraní narozdiel od predošlých hier. Typ zbrane sa vyberie ešte pred vložením kovovej tehličky do pece tým, že sa použije pečiatka s príslušným tvarom zbrane (obr. 3.4a). Síce má stále prednastavený počiatočný a konečný možný stav pre tvar zbraní (obr. 3.4b), možné je však prestať deformovať medzi týmito stavmi (obr. 3.4c, 3.4d). Okrem toho sa kovový materiál deformuje približne v mieste úderu kladiva. Je taktiež možné deformovať rôznou silou, tá určuje o koľko krokov sa posunie tvar medzi začiatočným a konečným stavom. Na záver stačí už len schlaďiť výrobok vo vode (obr. 3.4e), spojiť s násadami (obr. 3.4f) a zbraň je hotová.

Master Bladesmith

Hra Master Bladesmith je stále vo vývoji. Má podporu pre VR zariadenia Valve Index, HTC Vive, Oculus Rift a Windows Mixed Reality. Snaží sa o trochu realistickejšiu vizualizáciu oproti predošlým hrám a zároveň aj rozšíriť možnosti akcií.

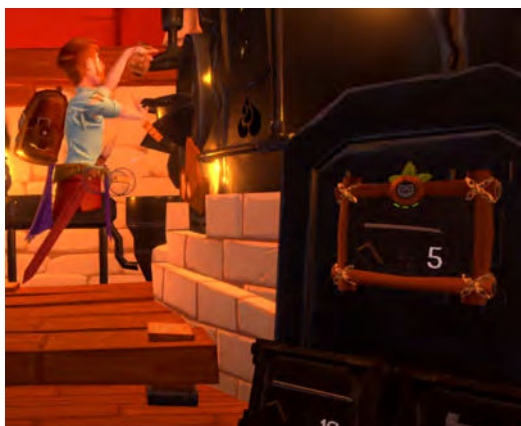
V tejto hre sa roztapia rudy v peci (obr. 3.5a) a odlejú do foriem na tehličky (obr. 3.5b). Tehličky sa potom dajú nahriať do pece aby sa z nich dalo kovať (obr. 3.5c). Typ výsledného produktu sa vyberá v knihe za nákovou. Potom sa pri udieraní kladivom na kovovú tehličku pomaly deformuje do tvaru vyobrazenom v knihe (obr. 3.5d). Pri dlhšom produkte je treba udierať do rôznych miest aby sa tá konkrétna časť predĺžila a dostala do finálneho stavu (obr. 3.5e). Aj táto hra sa teda drží predpripravených tvarov a ich stavov a kým sa objekt nedostane do finálneho stavu, tak sa z nej nedá vyrobiť výsledná zbraň. Po úspešnom ukončení kutia stačí kov schlaďiť (obr. 3.5f).

³<https://www.youtube.com/watch?v=I3p2DAyCyqY>, <https://youtu.be/ztzYnGaG91A>

⁴<https://www.youtube.com/watch?v=XA5ebuJtAwk>

⁵<https://youtu.be/9wKnygeGFw8>

⁶<https://www.youtube.com/watch?v=8tVVm7bFxsA>



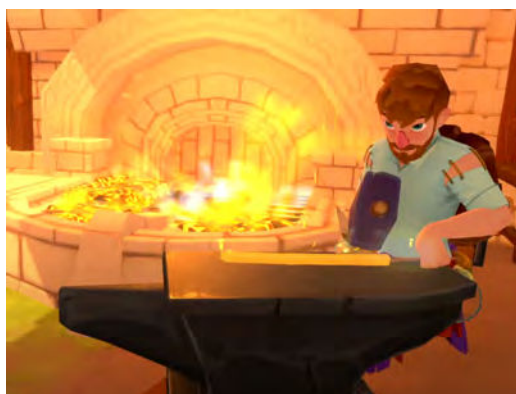
(a) Velká pec na spracovanie rúd, kovových tehličiek a nakoniec vyrobenie častí pre zbrane a nástroje. Vľavo vstupný priestor, vpravo šablóna pre výrobu konkrétneho tvaru.



(b) Výrobok z pece je hrubý, neopracovaný a nemá ostrie.



(c) Nahrievanie kovového odliatku v pahrebe



(d) Po pár úderoch sa hrubý odliatok rovno zmení na finálny naostrený výrobok



(e) Rôzne kovové výrobky je možné spojiť do zbraní

Obr. 3.2: Ukážka z hry Township Tale časť o kováčstve³



(a) Kovová ruda sa v peci zmení do tekutého stavu rovno v nádobe



(b) Tekutý kov sa odlieva do formy v tvare meča



(c) Po údere kladivom na formu kov stuhne a zmení sa do skoro finálnej podoby



(d) Po zmene môže byť meč vo výbornom stave alebo môže mať na sebe diery



(e) Dieravý meč sa vloží do pece kde sa zahreje a vyplnia sa medzery aspoň vizuálne

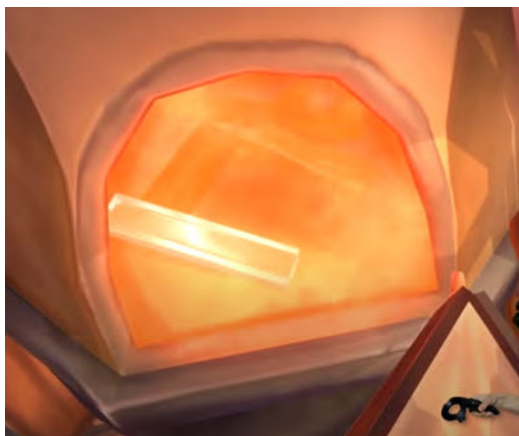


(f) Nahriaty meč sa po úderoch kladivom zmení do finálneho tvaru bez dier

Obr. 3.3: Ukážka z hry Craft Keep VR⁴



(a) Začína sa s kovovými tehličkami, ktoré sa opečiatkujú a tak sa určí typ výsledného produktu



(b) Po zahriatí kovovej tehličky v peci sa zmení do počiatočného stavu deformovateľného objektu podľa typu daného pečiatkou



(c) Nahriaty meč v základnom stave je možné deformovať na nákov. Deformuje sa približne podľa pozície a sily úderu o niekoľko krokov ku výslednému stavu



(d) Deformovať je možné hocikedy prestať, nemusí sa na všetkých miestach dostať do krajného stavu



(e) Na schladenie sa použije nádoba s vodou



(f) Ku ostriu meča je pridaná násada a drakom

Obr. 3.4: Ukážka z hry Fantasy Smith VR⁵



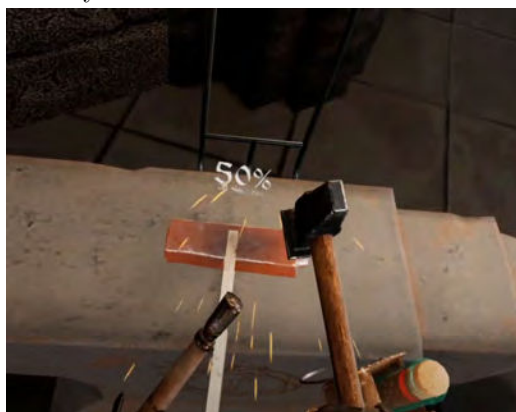
(a) Ruda sa roztápa v peci v nádobe



(b) Tekutý kov sa leje do väčšej formy v tvare tehličky



(c) Kovová tehlička je po vybratí z formy studená a treba ju nahriať v peci



(d) Nahriaty kov sa deformuje podľa toho na ktorú časť sa udiera kladivom



(e) Kovový objekt naberá tvar podľa predlohy v knihe



(f) Po úspešnom vytvorení ostria meča je schladený a môže sa použiť na zloženie meča

Obr. 3.5: Ukážka z hry Master Bladesmith⁶

Kapitola 4

Kováčstvo

Táto kapitola popisuje základné informácie o kováčskom remesle, ukážky potrebných nástrojov pre kovanie a rôznych kladív na deformáciu kovu. Venuje sa aj základným technikám, ktoré sú využívané pre deformáciu kovov za tepla.

4.1 Základy kováčstva

Kováčstvo je remeslom, ktoré využíva teplotu a mechanické spracúvanie kovov, najmä železa. Toto remeslo bolo v minulosti rozšírené pre jeho vyžitie v mnohých miestach, od väčších projektov na kostoloch, domoch a bránach, cez súčasti kočov a vozov, domáce využitie výrobkov do poľa či do kuchyne, až ku ornamentálnym kúskom. Dnes je toto remeslo menej rozšírené, hlavne kvôli strojom, ktoré dokážu spracovať a upraviť kovy do požadovaných tvarov, ale aj kvôli iným materiálom, ktoré nahrádzajú kovy (adaptované z encyklopédie ÚLUV, časť Kováčstvo [11]).

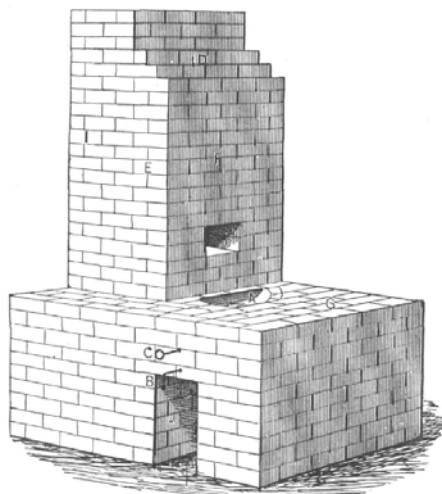
Kovať je možné za studena i za tepla (z článku Nahrman a Matzenmiller [13]), pričom za studena je možné kovať len mäkkšie kovy, ako napríklad cín a meď (ako je uvedené v príručke *Hot forging and cold forging* [1]). Kovanie za tepla je viac používané, keďže železo, jedno z najviac používaných kovov, je možné kovať iba za dostatočne vysokej teploty [11].

V minulosti bolo treba veľa času stráviť iba prípravou tohto kovu z nečistých hrúd. Z neho potom bola vytavená zliatina, ktorá mohla byť kovaná do výsledného diela. V súčasnej dobe je možné kúpiť už predpripravené kusy kovového materiálu rôznych parametrov podľa potrieb pre výsledný produkt.

4.2 Nástroje

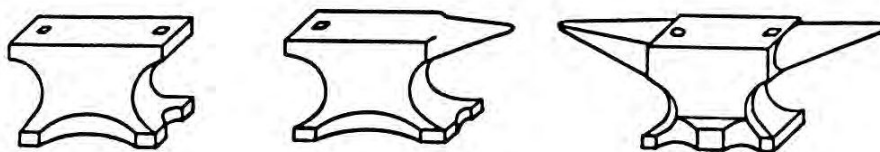
Pre kovanie za tepla sú potrebné okrem iných, tri základné nástroje: vyhňa, nákova a kladivo. Vyhňa zabezpečuje teplo, ktoré zmäkčuje kov a nákova a kladivo sú oboje nástroje, ktorými sa kov tvaruje [11].

K vyhni (obr. 8.2, prevzaté z knihy *Practical Backsmithing* [14]) patria taktiež mechy, v tejto dobe aj ventilátory, ktoré vháňajú vzduch do vyhne podľa potreby a tým rýchlo upravujú intenzitu tepla [11].



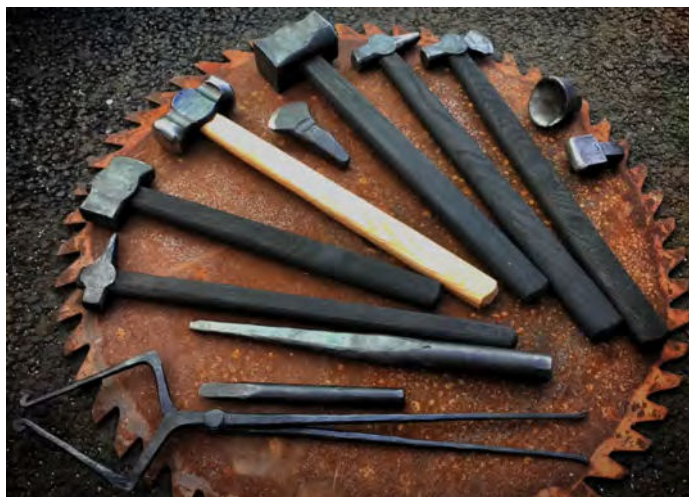
Obr. 4.1: Príklad vyhne používanej pri kovaní za tepla

Nákova je taktiež dôležitý predmet pre kováča, ktorý slúži ako podstava a zároveň podložka pre kovaný predmet a taktiež ako narážka pre údery kladiva. Tvary nákovy sa líšia medzi bezrohé, jednorohé a dvojrohé (obr. 4.2, prevzaté z encyklopédie: Tradičné remeslá a domáce výroby, časť o Kováčstve [11]). Väčšina nákov ale majú spoločné dva otvory, jeden kruhový a jeden štvorcový, ktoré slúžia na vkladanie pomocných nástrojov, používané napríklad na pridŕžanie či ohýbanie [11].



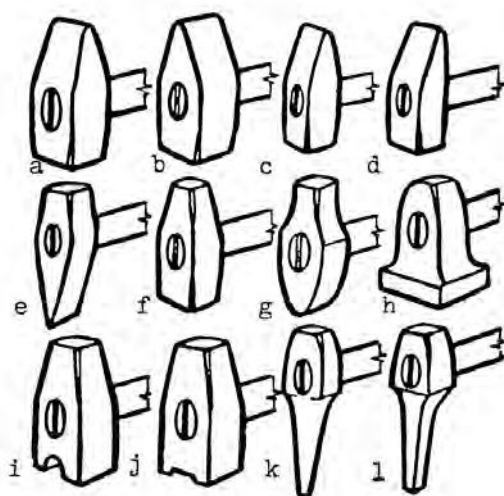
Obr. 4.2: Tri druhy nákov: bezrohá, jednorohá, dvojrohá (z ľava do prava)

Kladivá majú taktiež mnohé tvary a využitia (obr. 4.4, prevzaté z encyklopédie, časť o Kováčstve [11]). Rôzne kladivá majú rôzne funkcie, napríklad, na rozširovanie, sekanie či predlžovanie.



Obr. 4.3: Prehľad kováčskych kladív a nástrojov určených pre deformáciu kovov¹. V strede kladivo s oblou časťou na jednej strane a plochou na druhej.

Kladivá s jednou alebo oboma úzkymi stranami rozširujú kov do jednej strany, podľa toho, v akom uhle kováč kov udrie. Kladivá so širokým oblým koncom (možné vidieť na obrázku 4.3), ktorý vytvára okrúhlu indentáciu, pôsobia na kov do všetkých strán rovnakou silou a tak je možné rozšíriť kov do všetkých strán naraz (adaptované z knihy *Practical Blacksmithing* [14]).



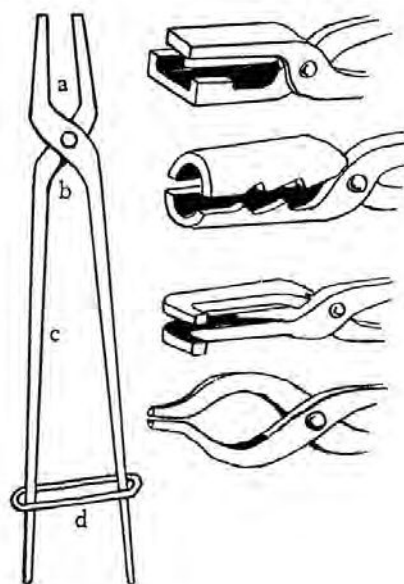
Obr. 4.4: a) dvojručné s rovnobežným nosom, b) dvojručné s priečnym nosom, c) jednoručné s rovnobežným nosom, d) dvojručné s rovnobežným nosom (ľahšie), e) sekáč, f) široký sedlík, g) úzky sedlík, h) hladiaci sedlík, i) zápustkové kladivo s valcovou dutinou, j) zápustkové kladivo s hranatou dutinou, k) prebíjacie kladivo valcové, l) prebíjacie kladivo hranaté

Samozrejme, kováči používajú mnohé kladivá, ako je aj prezentované na obrázku 4.4, ktoré sú používané na špecializované práce. Príkladom takýchto špecializovaných kladív je prebíjacie kladivo (kladivá k a l), ktoré sa používa na vytvorenie diery.

¹<https://www.youtube.com/watch?v=eIn9aVbdF1Y>

Podľa knihy *Cognition and tool use: The blacksmith at work* [10] (informácie z knihy adaptované pre tento paragraf), kov je taktiež pri kovaní ovplyvňovaný nákovou. Používa sa ako oporná podložka pri kovaní kladivom, ktorá zabráňuje pohybu kovu. To však nie je jej jediná funkcionálnosť. Môžu sa využívať jej hrany, schody, roh a diery či už ako prostriedky na detailné deformovanie kovu samotného ale aj ako držiak pre ďalšie nástroje na úpravu kovov. Náková teda môže poslúžiť akoby druhé kladivo a tak urýchliť prácu deformácie kovu. Náková sa navyše pri náraze neposúva a deformuje presne tú časť kovu, ktorá k nej bola priložená. To znamená, že to je presnejšie než úder pohyblivého kladiva. Roh nákovy sa využíva na ohýbanie kovu, čo je náročnejšie dosiahnuť samotným kladivom na plochej časti nákovy. Pre typ a tvar deformácie záleží taktiež na uhle a priložení kovového materiálu o nákovu (na obrázkoch 4.7, kov priložený pod uhlom a kladivo udiera smerom na nákovu vľavo, mimo nákovy vpravo a 6.9, kov priložený v 45° ku povrchu nákovy na jej hrane, kladivo udiera kolmo na povrch kovu zrkadlovo v mieste priloženia kovu ku nákove).

Okrem kladiva a nákovy, kliešte sú taktiež veľmi dôležité a pomáhajú kováčovi udržiavať kov na mieste pri opracovávaní. Kliešte majú rôzne tvary, ktoré slúžia na uchopenie kovov s rôznymi veľkosťami a tvarmi. Napríklad, skrinkové kliešte sú používané na uchopenie plochého podlhovastého kovu, kým kliešte nazývané vlčie zuby dokážu udržať mnohé tvary, vďaka zubom, ktoré zabránia klzaniu. Obe tieto kliešte sú vyobrazené na obrázku 4.5 (prevzaté z encyklopédie, časť o Kováčstve [11]). Pre kratšie časti kovu sa často používa predĺženie v podobe kovovej tyče na uchopenie kovu pri zahrievaní a deformácii. V prípade, že kov je príliš krátky, je možné použiť na uchytenie kliešte namiesto kovovej tyče.

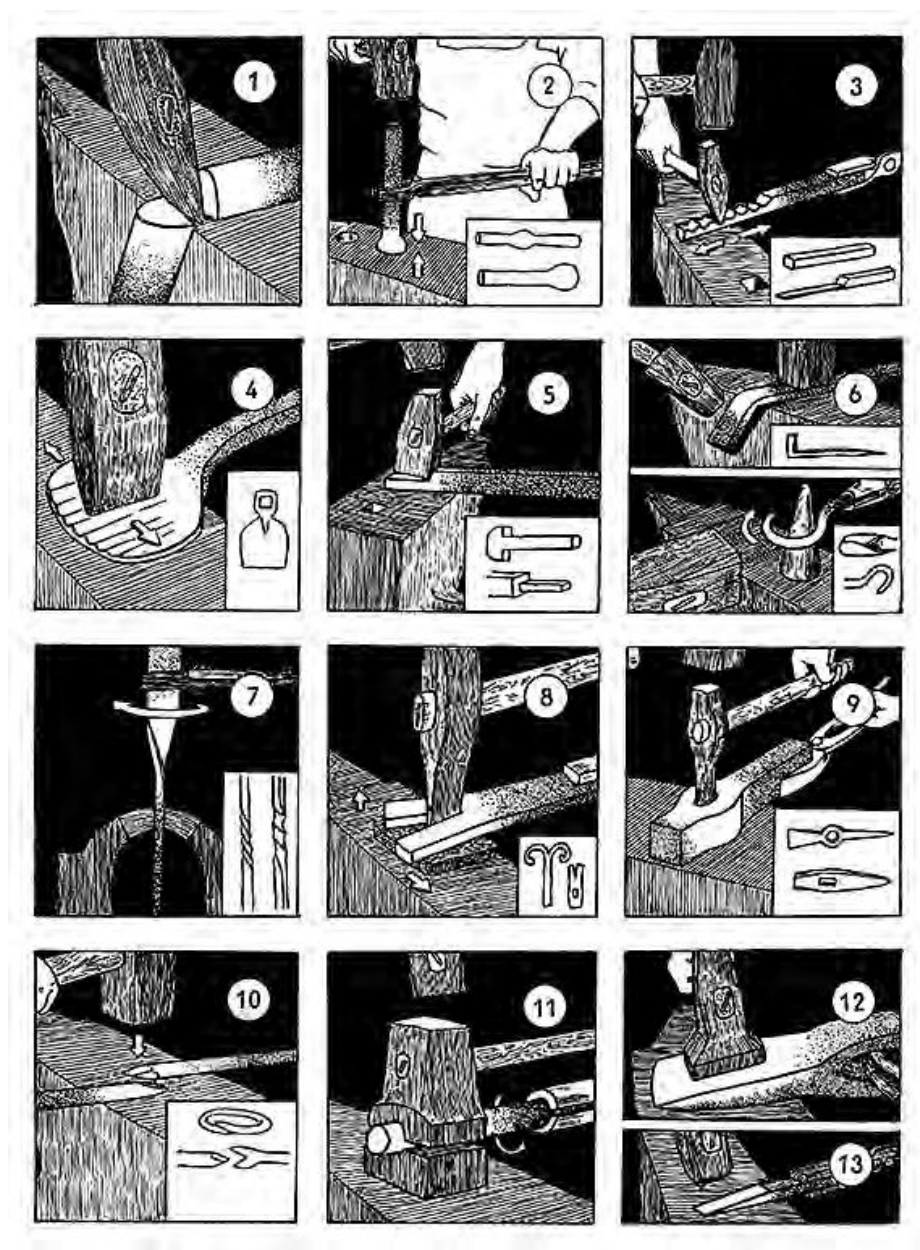


Obr. 4.5: Ukážka klieští rôznych druhov: skrinkové kliešte, vlčie zuby, dvoje kliešte na uchopenie tenkých predmetov (z hora nadol)

4.3 Techniky

Základná technika pre každého kováča je ručné kovanie kladivom za tepla. Kováč musí odhadnúť teplotu kovu podľa jeho sfarbenia po nahriatí aby mohol kov riadne spracovať.

Technicky zdatný kováč dokáže kov deformovať pomocou rôznych techník, ktoré mu umožnia dopracovať sa k výslednému produktu. Striedanie a miešanie týchto techník mu pomôže tvarovať kov do konečného výsledku, ktorý by bol ťažko dosiahnuteľný ak by použil len jednu techniku.



Obr. 4.6: Techniky kovania za tepla: 1) sekanie, 2) zrážanie, 3) vyťahovanie, 4) rozširovanie, 5) osadzovanie, 6) ohýbanie do uhla a do oblúka, 7) krútenie, 8) štípenie, 9) prebíjanie, 10) kováčske zváranie, 11) zápustkové kovanie, 12-13) ostrenie

Na obrázku 4.6 (prevzaté z encyklopédie, časť o Kováčstve [11]) sú vyobrazené niektoré základné techniky kovania, ktoré by každý zručný kováč mal ovládať. Aj tu je možno vidieť rozdielne kladivá, ktoré sú určené na jednotlivé techniky.

Kováčske techniky môžeme rozdeliť na techniky udierania kladivom, techniky kovania, techniky zvárania a techniky tepelného opracovania.

Techniky udierania kladivom závisia od toho, na aké miesto sa kladivom udiera, v akom uhle sa udiera a taktiež v akom uhle sa drží železo. Nasledujúce techniky sú adaptované a voľne preložené z Kováčskej lekcie 5, v *The Eight Basic Hammer Strikes* od Kalif [8]. Môže sa to zdať jednoduché, ale treba vziať do úvahy, že pri práci s nahriatym železom sa neupravuje iba kladivom ale aj nákovou. Medzi základné techniky udierania kladivom patria:



Obr. 4.7: Angle Hammer (naľavo), Backface Shearing (napravo)²

- Paralelné - udiera sa pri ňom kladivom paralelne na plochu nákovy
- Pod uhlom - kladivo udiera na železo pod uhlom voči ploche nákovy
- Polovica kladiva blízko - kladivo udiera polovicou hlavy na kovadlinu a polovica je mimo nákovy, kladivo teda udiera na bližšiu hranu plochy nákovy
- Polovica kladiva ďaleko - úder kladiva je mierený na vzdialenú hranu plochy nákovy, tak aby sa polovica hlavy kladiva nachádzala nad nákovou a druhá polovica mimo nákovy
- Hrana kladiva blízko - v tom to prípade je celá hlava kladiva nad nákovou ale okraj hlavy kladiva je zarovnaný s blízkou hranou nákovy
- Hrana kladiva ďaleko - podobne je kladivo nad nákovou a hrana hlavy kladiva je zarovnaná tento krát so vzdialenou hranou plochy nákovy
- Zadná strana - hlava kladiva udiera do železa vytrčajúceho z nákovy v spätnom údere smerom nazad ku hrane nákovy (vzniká tak takzvaný upsetting alebo osadzovanie na železe).
- Strihanie - celá hlava kladiva je mimo plochy nákovy a udiera do železa, ktorá z nákovy pretŕča

²Prevzaté z <https://web.archive.org/web/20210613051444/https://stormthecastle.com/blacksmithing/blacksmithing-lessons/blacksmithing-lesson-5-the-8-basic-hammering-strikes.htm>

Techniky kovania (forging) zahŕňajú použitie rôznych kladív a typov úderov kladivom na rozohriate železo umiestnené na nákovu. Delia sa podľa efektu, ktorý majú na železo. Medzi základné techniky kovania patria vyťahovanie a rozširovanie, ohýbanie, osadzovanie a prebíjanie. Tieto techniky sú zväčša opísané a voľne adaptované v nasledujúcej „časti“ podľa knihy *Basic Blacksmithing* [7].

Vyťahovanie a rozširovanie (drawing) železo stenšuje ale zároveň predlžuje alebo rozširuje. Pre túto metódu je potrebné nahriať železo na teplotu určenú pre kovanie. Vizualne je možné túto teplotu rozoznať ako jasne červenú žiaru. Železo je možné držať rukou v prípade, že je materiál dostatočne dlhý a na konci chladný na dotyk, inak sa drží pomocou klieští. Nahriate železo sa položí na nákovu a kladivom sa udiera tak, aby plochá časť (hlava) kladiva mierila priamo na železo. Pri údere sa železo hýbe v smere od stredu plochy kladiva do ostatných smerov. Rozsah a smer šírenia závisí od úderu a tvaru kladiva. Kladivom sa udiera tak dlho, kým nie je dielo dostatočne tenké, tak ako je požadované, alebo kým je kov dostatočne teplý. Udieranie do chladného kovu by ho mohlo zničiť, a preto je nutné ho znovu nahriať a opakovať proces kovania. Dobré je naplánovať si celý proces tak, aby boli náradia jednoducho dostupné, a minimalizovať vzdialenosť medzi nahrievaním kovu v peci a nákovou, pretože prenos materiálu je stratený čas. Pri pravidelnej práci je tak možné nájsť spôsoby na zvýšenie produkcie [7].

Smer do ktorej sa kov posúva (tlačí) pôsobením úderu kladiva a nákovy z opačnej strany je možné usmerňovať viacerými možnosťami. Je možné nahriať iba časť kovu, ktorá sa má upravovať alebo ochladiť tú časť, ktorá nemá byť upravená. Na najväčšie posúvanie masy železa sa používajú konkrétne typy kladív a to zúžené ku koncu (napríklad cross-peen). Pomocou tohto typu kladiva sa urobí séria priehlbín v požadovanom smere a tie sú potom vyrovnané pomocou plochej strany kladiva [7].

Ohýbanie (bending) je technika, pri ktorej sa nahreje na teplotu kovania tá časť kovu, ktorá má byť ohýbaná. Jeden koniec železa je uchytený pomocou klieští alebo iného nástroja (ohýbacia vidlica, zverák) a druhý koniec sa otáča. Používa sa taktiež ohýbanie cez okraj nákovy alebo na rohu nákovy pomocou úderov kladivom [7].

Osadzovanie (upsetting) je technika opačná ku vyťahovaniu a rozširovaniu, teda kov bude hrubší, ale kratší alebo užší. Táto metóda je náročnejšia a ťažšie sa ovláda, preto sa nepoužíva až tak často. Pri tejto metóde sa upravovaná časť nahreje, jeden koniec kovu sa umiestni proti nákovu a na druhý sa udiera smerom ku nákovu. Tak vznikne zhrubnutejšia časť v oblasti kde bol kov nahriaty [7].

Prebíjanie (punching) sa používa ak je treba do kovu urobiť dieru. Nahriaty kov sa položí nad štvorcový (väčší) otvor nákovy, nástroj na prerazenie diery sa opatrne umiestni na kov v mieste, kde má vzniknúť diera a nakoniec sa na nástroj prudko udrie. Nástroje na vytvorenie diery môžu mať rôzne tvary podľa požadovaného výsledku. Z dôvodu nutnosti použitia viacerých nástrojov, je vhodné na uchytenie kovu na mieste použiť pridržovacie nástroje. Môže byť použitá hrubá zatažená reťaz alebo nástroje umiestnené do okrúhleho (menšieho) otvoru [7].

Zváranie (welding) je metóda, ktorá spája kusy kovu do jedného. Táto metóda je pomerne náročná a tak sa jej mnoho kováčov radšej vyhýba. V dávnejších dobách sa využívala hlavne na vytvorenie základného kovu na ďalšie opracovávanie, pretože nebolo možné získať čistý kus kovu. Kov sa získaval z rôznych nečistých hrúd, tie sa prekladali a zvárali dokopy aby sa na povrch dostali nečistoty a mohli byť odstránené, a tak bol kov pripravený na kovanie produktu (adaptované z článku od Krastojković a Bajić [9]).

V súčasnosti už nie je nutné vyrábať čistý kov, keďže sa dá získať jednoduchšie a začať tak rovno kovať. Zvárajú sa teda časti už obrábaného kovu. Kusy kovu, ktoré majú byť

spojené musia byť dostatočne očistené aby ich bolo možné spojiť Táto metóda vyžaduje nahriatie kovu na teplotu vhodnú na zváranie. Diely na spojenie treba držať dostatočne blízko seba, to je možné dosiahnuť nitovaním kusov k sebe, alebo dočasným zvarením. Nakoniec je potreba ich kovať dokopy rôznymi údermi aby sa spojili [7].

Kov sa na konci práce tepelne upravuje aby sa zvýšila tvrdosť alebo naopak ohybnosť. Môže sa použiť nádoba s vodou do ktorej sa dielo ponorí, alebo olej ktorý ochladzuje dielo pomalšie. Ďalej sa využíva popol, perlit a pomalé ochladzovanie zmenšovaním ohňa. Pomocou práškového uhlia sa môže dosiahnuť zvýšenie obsahu uhlíka na povrchu zliatiny s relatívne nízkym obsahom uhlíka [7].

Kapitola 5

Technológie

Táto kapitola rozoberá technológie, ktoré bolo potrebné nastudovať pre vypracovanie diplomovej práce. Konkrétne sa jedná najmä o Unity 3D engine, Virtuálnu realitu a aplikáciu Blender, využívanú na vytváranie 3D objektov a textúr. Z aplikácií sú popísané predovšetkým časti najviac využiteľné pre daný typ zadania práce.

5.1 Unity 3D

Podľa článku od Xie [23], Unity 3D je multiplatformný herný engine od spoločnosti Unity Technologies. Vyše dvadsať rokov vývoja umožnilo prejsť z jednoduchých 2D projektov na možnosť vytvárania 3D modelov a animácií. V súčasnosti podporuje viac ako 25 platforiem. Je vhodný na vytváranie 2D, 3D aplikácií využívajúcich virtuálnu realitu (VR) aj rozšírenú realitu (AR) a simuláciu. Používa sa na vytváranie hier, filmov a animácií, v automobilovom priemysle, architektúre, strojárstve aj stavebníctve. Jeho výhodou je, že namiesto dlhých komplikovaných kódov, vývojári (developeri) dokážu vytvoriť komplexné scény pomocou vizuálneho prostredia editora za pomoci základnej stavebnej jednotky, ktorou je GameObject. Ten je možné jednoducho nastavovať a upravovať pomocou konfigurácie jednotlivých komponent za pomoci užívateľského rozhrania (UI).

Primárnymi skriptovacími jazykmi sú C# a Javascript, i keď v starších verziách bolo možné použiť aj jazyk Boo (z článku od Buyuksalih a kol. [5]). Avšak od verzie 2017.1 sa využíva hlavne C#. Najnovšie je možné použiť aj vizuálne skriptovanie pomocou oficiálneho rozšírenia Bolt (z Unity manuálu [20]).

Unity obsahuje a podporuje nasledujúce funkcionality [20]:

- 2D,
- Grafika,
- Fyzika,
- Sieť,
- Scriptovanie,
- Zvuk,
- Animácia,
- UI,
- Navigácia,
- Editor,
- Asset Store.

Grafika

Táto časť zahŕňa osvetlenie, kamery, materiály, tieňovače (shaders), textúry, častice (particles) a vizuálne efekty. Informácie pre túto sekciu boli adaptované a čiastočne voľne preložené z Unity manuálu, časti *Unity User Manual 2020.2* [20] a *Graphics* [19].

Osvetlenie v Unity funguje tak, že aproximuje, ako sa svetlo správa v skutočnom svete. Pre dosiahnutie realistickejšieho, zjednodušeného alebo štylizovaného výsledku používa rôzne podrobné osvetľovacie modely. Používa priame a nepriame osvetlenie, real-time a baked osvetlenie, globálne osvetlenie, baked globálne osvetlenie, real-time globálne osvetlenie. Priame svetlo je také, ktoré je emitované, raz dopadne na povrch a odrazí sa od povrchu priamo do senzora (môže ním byť kamera alebo sietnica oka). Nepriame svetlo zahŕňa všetky ostatné svetlá odrážané do senzora, teda aj tie ktoré sa odrážajú viackrát a svetlo oblohy. Pre vytvorenie realistického osvetlenia je potrebné použiť (simulovať) ako priame, tak aj nepriame osvetlenie. Real-time osvetlenie je osvetlenie v reálnom čase. To znamená, že Unity počíta osvetlenie za behu programu. Baked osvetlenie naopak počíta pred behom aplikácie. Výsledky sú potom uložené ako svetelné údaje, neskôr použité pri behu aplikácie. V Unity je možné použiť aj kombináciu týchto osvetlení. Globálne osvetlenie je skupina techník, ktoré modelujú priame a nepriame osvetlenie pre vznik realistických výsledkov osvetlenia. Unity má dva takéto systémy pre kombináciu. Systém baked globálne osvetlenie obsahuje svetelné mapy, svetelné sondy a reflexné sondy. Real-time globálne osvetlenie je možné použiť pomocou vstavanej rendrovacej pipeline. V blízkej dobe bude tento systém z Unity odstránený.

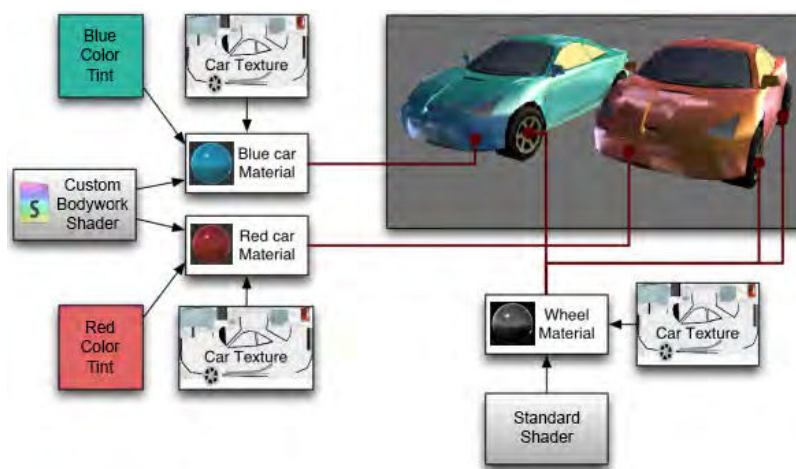
Kamera je reprezentovaná objektom v scéne GameObject ako aj ostatné typy objektov. To čo má kamera v zornom poli je definované jej transformáciou a nastaveniami. Hľadisko kamery je určené polohou transformácie, osa vpred (Z) definuje smer pohľadu a osa nahor (Y) definuje hornú časť obrazovky. Nastavenia kamery ďalej definujú veľkosť a tvar oblasti, ktorá spadá do zobrazenia. Projekcie kamery podporované v Unity sú perspektívna a ortogonálna (obr. 5.1). Kamera orezáva objekty ktoré sú príliš blízko alebo ďaleko od kamery podľa Z - osi od kamery (tzv. blízka a ďaleká orezávacía rovina). Medzi týmito rovinami je viditeľný rozsah kamery. Ďalej sa orezávajú okraje buď pomocou obdĺžnika pri ortogonálnom premietaní alebo podľa pyramídy s vrcholom v polohe kamery a základňou v ďalekej orezávacej rovine pri perspektívnom premietaní. Kvôli blízkej orezávacej rovine tento tvar nie je úplná pyramída, ale zrezaná pyramída nazývaná frustum. Kamere sa tiež nastavuje pozadie, ktoré je vidieť medzi objektami. Môže to byť farba, obloha alebo vzdialené pozadie, poprípade obsah poslednej snímky kamery.



Obr. 5.1: Perspektívne (vľavo) a ortografické (vpravo) zobrazenie tej istej scény¹

¹<https://docs.unity3d.com/2020.1/Documentation/Manual/CamerasOverview.html>

Pre renderovanie v Unity sa využívajú meshe, materiály, shadery a textúry (obr. 5.2). Meshe sú hlavným grafickým primitívom v Unity. Definujú tvar objektu. Materiál definuje ako sa má povrch objektu vykresliť. Obsahuje odkazy na textúry a rôzne nastavenia. Tie závisia od toho aký shader je použitý pre konkrétny materiál. Shadery sú malé skripty obsahujúce matematické výpočty a algoritmy na výpočet farby každého pixelu vykresleného na základe vstupného osvetlenia a konfigurácie materiálu. Pod textúrami sa rozumejú bitmapové obrázky. Materiál môže obsahovať odkazy na viacero týchto textúr, ktoré sú použité pri výpočtoch v shaderoch. V Unity je možné použiť niekoľko preddefinovaných shaderov, ako napríklad štandardný shader. Ten je najčastejšie vhodný použiť pre bežné vykreslenie postáv, scény, prostredia, pevných a priehľadných objektov, tvrdé aj mäkké povrchy. Pre vykreslenie kvapalín, lístia, skla, časticových efektov, umelecké alebo iné špeciálne efekty je možné použiť iný vstavaný shader, alebo vytvoriť vlastný.



Obr. 5.2: Ukážka diagramu využívajúceho 3 materiály, 2 shadery a 1 textúru pre vyobrazenie modelov²

Časticové systémy simulujú a vykresľujú veľa malých obrázkov alebo meshov, nazývaných častice, aby vytvorili vizuálne efekty. Systém simuluje každú časticu kolektívne, aby vytvoril dojem úplného efektu. Systém častíc je užitočný pri vytváraní dynamických objektov, ako je napríklad oheň, dym alebo kvapaliny, pretože je náročné zobrazit tento druh objektov pomocou meshu (obr. 5.3).



Obr. 5.3: Holo stôl vytvorený pomocou časticového systému³

²<https://docs.unity3d.com/520/Documentation/Manual/Materials.html>

Fyzika

Unity umožňuje simulovať fyziku v projekte pre zabezpečenie akcelerácie, reakcie na kolízie, ovplyvnenia objektu gravitáciou a ďalšie sily. Je možné využiť vstavaný fyzikálny engine podľa potreby. Pre 3D je zabudovaná fyzika integráciou engine Nvidia PhysX a pre 2D zas integrácia engine Box2D. Pre dátovo orientované projekty je možné využiť rôzne balíky fyzikálnych enginov. Sú to napríklad: balík Unity Physics (fyzikálny engine DOTS), balík Havok Physics pre Unity (rozšírenie Unity Physics). [20] z časti *Unity User Manual*⁴, [18] z časti *3D Physics for object-oriented projects*⁵.

Hlavné koncepty 3D fyziky v Unity sú: Rigidbody, Colliders, Joints, Character Controllers [20] z časti *Unity User Manual* a [18] z časti *3D Physics for object-oriented projects*.

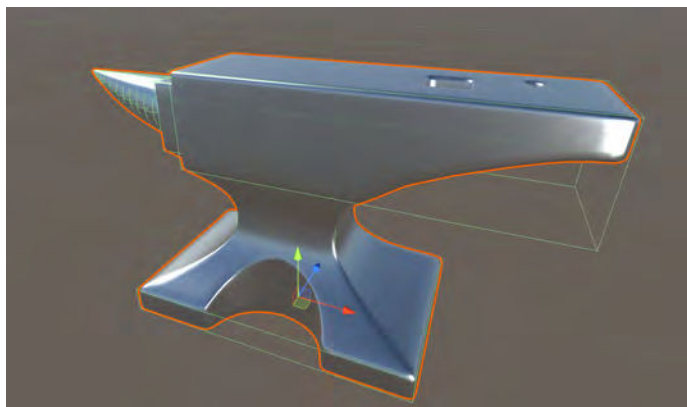
Rigidbody je hlavná komponenta umožňujúca fyzikálne správanie objektu GameObject. Po pridaní tejto komponenty, objekt automaticky reaguje na gravitáciu. Ak je objektu pridaná aj komponenta collider, objekt pri kolízii je posúvaný na základe prichádzajúcich kolízií. Použitím rigidbody sa preberá kontrola nad pohybom objektu a tak je vhodnejšie takýto objekt pohybovať nie zmenou transformácie, ale pôsobením síl. Tie fyzikálny engine prepočítava na pohyb. V prípade, že objekt nemá byť riadený enginom ale na základe vstupov od zariadenia ako napríklad controller (ovládač), ale zároveň zachovať možnosť detekcie triggerami. Tento druh nefyzikálneho pohybu sa nazýva kinematický. V komponente rigidbody je možné nastaviť vlastnosť Is Kinematic. Táto vlastnosť vypína riadenie fyzikálnym enginom a umožňuje manipuláciu pomocou scriptu. Pri pohybe objektu engine šetrí výpočetný čas procesora tak, že pri veľmi pomalom pohybe objektu usúdi že objekt stojí na mieste a uspí ho. Tento objekt je znova prebudený pri kolízii, ktorá ho ovplyvní a začne posúvať. V prípade núdze je možné prebudenie explicitne a to funkciou WakeUp [20] z časti *Unity User Manual*, [18] z časti *3D Physics for object-oriented projects*.

Collider (obr. 5.4) sú komponenty, ktoré definujú tvar GameObjectu na účely detekcie fyzikálnych kolízií. Collider sú neviditeľné a nemusia mať presný tvar ako je mesh objektu. Často sa používa hrubá aproximácia meshu, lebo je efektívnejšia a často to je na nerozoznanie za behu aplikácie. Najjednoduchšie a zároveň najmenej náročné pre procesor sú primitívne typy Colliderov. V 3D sú to Box Collider, Sphere Collider a Capsule Collider (podobne aj v 2D). Na jeden GameObject je možné pridať viac colliderov.

³<https://docs.unity3d.com/2021.1/Documentation/Manual/ParticleSystem.html>

⁴<https://docs.unity.cn/2020.2/Documentation/Manual/UnityManual.html>

⁵<https://docs.unity3d.com/2020.2/Documentation/Manual/PhysicsOverview.html>



Obr. 5.4: Ukážka kombinácie jednoduchých colliderov na nákovce, box collider tvaru v tele nákovce a sphere collider guľového tvaru v rohu nákovce

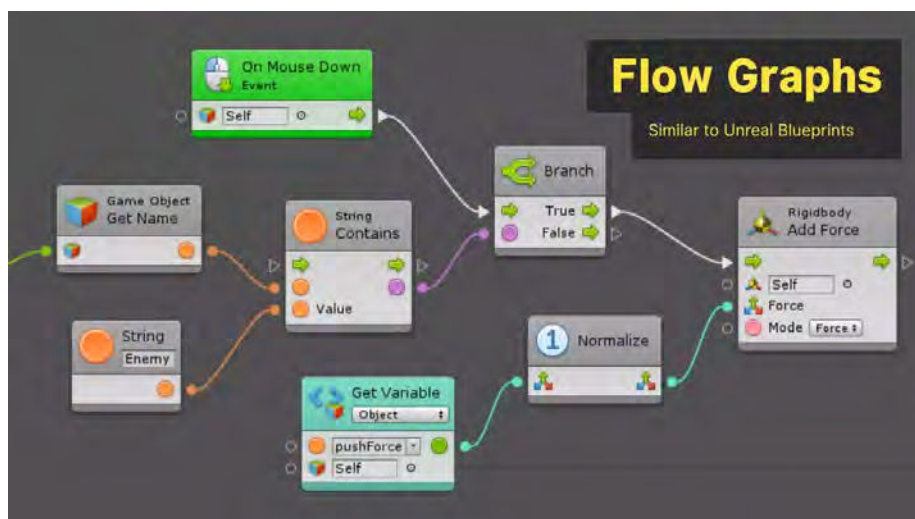
Pre presnejšiu reprezentáciu objektu je možné použiť zložené (compound) collider, stále majú nižšiu náročnosť na procesor. V niektorých prípadoch, kedy ani tieto collider nie sú dostatočné je možné použiť mesh collider. Mesh collider presne zodpovedajú tvaru meshu objektu. Nevýhodou je, že sú oveľa náročnejšie na procesor než primitívne typy. V prípade kolízie dvoch objektov s mesh colliderami sa pri kontakte nič nestane, lebo jeden mesh collider nemôže naraziť do iného. Niekedy sa to rieši tak, že sa collider označí ako konvexný. Vtedy je vytvorená konvexná obálka pre daný mesh objektu, takže všetky priehlbiny a diery sú prekryté. Tento typ sa nastavuje, pretože sa konvexný collider môže zraziť s iným mesh colliderom. Pravidlom je že mesh collider by sa mal používať iba na scénu a pre zložitejšie objekty použiť zložené collider. Pre nehybné objekty ako steny a podlahy je možné použiť statické collider. Tieto collider je možné použiť aj bez rigidbody. Dokážu kolidovať s dynamickými colliderami, ale pri zrážkach sa nepohybujú. Ak nie je potrebné zisťovať kolízie oboch objektov a následné ovplyvnenie pohybu kvôli kolízii, ale iba zistiť či sa jeden kolízny objekt dotkol alebo prešiel cez iný, používajú sa trigger. Objekt označený ako trigger sa nespráva ako solídny objekt, ale umožňuje ostatným objektom cez neho prechádzať [20] [18], z časti *3D Physics for object-oriented projects*.

Joints (klíby) sa používajú na spájanie jedného pevného telesa s iným pevným telesom. Tieto spojenia môžu byť rôznych typov a používať rôzne sily, sú ale obmedzované limitami klíbov. Používajú sa character joint, configurable joint, fixed joint, hinge joint a spring joint [20] [18], z časti *3D Physics for object-oriented projects*.

Skriptovanie

Skriptovanie je podstatnou súčasťou Unity. Unity manuál časť *Unity engine visual scripting* [22] popisuje skriptovanie ako umožňujúce upravovanie väčšiny vecí ako pohybu, komponentov, materiálov a omnoho viac. Dáva voľnosť developerom nie len spravovať nastavenia ale aj vytvárať vlastné veci podľa predstavivosti. V súčasnosti sa pre skriptovanie (programovanie) používa hlavne jazyk C#. V starších verziách Unity boli podporované aj jazyky UnityScript (na štýl JavaScriptu) a jazyk Boo. Nasledujúca časť adaptuje a čiastočne voľne prekladá unity manuál časti *Unity User Manual 2020.2*. [20] a *Unity engine visual scripting* [22].

Novinkou je ale jazyk Bolt, ktorý sa používa na vizuálne skriptovanie. Namiesto iba písania riadkov kódu je Bolt založený na vizuálnom systéme s využitím grafov. Bolt ponúka vývojové grafy (obr. 5.5), stavové grafy, úpravy za behu programu, prediktívne ladenie a analýzu, kompatibilitu s Unity metódami, vlastnosťami aj skriptami samotnými. Vývojové grafy sú hlavné nástroje pre vytváranie interakcie s projektom. Pomocou akcií a hodnôt v uzloch je možné vykonávať logiku v ľubovoľnom poradí, buď v každej snímke alebo pri konkrétnej udalosti.



Obr. 5.5: Bolt vývojový (flow) graf je nástroj na vytváranie interakcie s projektom, lepšiu vizualizáciu aj za behu programu vrátane možných úprav za behu programu bez nutnosti znovu prekladať kód⁶

Stavové grafy umožňujú vytvárať samostatné správanie, ktoré určia objektom aké sekcie majú vykonať v konkrétnych stavoch. Sú vhodné pre logiku na vyššej úrovni, ako je správanie AI, štruktúru scény alebo levelu (úrovne), poprípade iného aspektu scény vyžadujúceho správanie, ktoré prechádza medzi stavmi. Živé úpravy sú nezvyčajné, lebo pri normálnom skriptovaní alebo konfigurácii nastavení za behu programu sa úpravy neukladajú. V tomto prípade sa dajú vykonávať zmeny aj po spustení v režime Play. Vďaka tomu je možné vidieť automaticky úpravy vizuálneho skriptovania za behu aplikácie (obr. 5.6). Tento spôsob je omnoho rýchlejší, lebo nie je potrebné znovu kompilovať zmeny projektu.

⁶<https://assetstore.unity.com/packages/tools/visual-scripting/bolt-163802>



Obr. 5.6: Bolt graf za behu programu pri prenose dát a volaní funkcií blokov

Medzi uzlami je možné za behu programu vidieť aktuálnu prenášanú hodnotu a tak je jednoduchšie debugovanie. Dokáže zistiť chyby za behu programu, pred spustením programu, nulové referencie a upozornenie na nebezpečné uzly, ktoré môžu mať najskôr problém.

Zvuk

Systém Unity poskytuje rôzne spôsoby zakomponovania zvuku do tvorenej aplikácie. V snahe napodobniť realitu, systém definuje entity, ktoré zvuk emitujú a prijímajú, pričom sa berie do úvahy ich poloha v priestore. Pre simulovanie efektu pozície musia byť zvuky naviazané na komponenty **AudioSource**, ktoré sú súčasťou herného objektu, ktorý predstavuje zdroj tohto zvuku. Vysielané zvuky z rôznych zdrojov následne zachytáva komponent **AudioListener**⁷ naviazaný k ďalšiemu hernému objektu, najčastejšie sa jedná o hlavný kamerový objekt scény. Unity dokáže simulovať efekt vzdialenosti zdroja zvuku a pozície od poslucháča a prehrať výsledný upravený zvuk v závislosti od týchto premenných do výstupného zariadenia. Zvukové komponenty obsahujú rôzne nastavenia pre ovplyvnenie výsledného výstupu.⁸

Unity dokáže pracovať s rôznymi druhmi zvukových stôp, vrátane bežných formátov ako mp3, wav a ogg. Pridaním do stopy do projektu sa zo zvukového assetu stane objekt **AudioClip**, ktorý sa potom vkladá do komponentu **AudioSource** cez Unity Editor, prípadne programátorsky v skripte. Prehrávaný klip zo zdroja je podľa potreby vývojára nakonfigurovaný a ovplyvňovaný týmto komponentom.

Príklad konfigurovateľných parametrov:⁹

- Play on Awake – automatické prehratie zvuku pri spustení scény a aplikácie.
- Loop – automatické znovuprehrávanie v cykle
- Volume – Nastavenie ako hlasno bude zvuk prehrávaný z jednotkovej vzdialenosti od objektu s **AudioListener**
- Spatial Blend – Upravuje mieru 3D efektu zvuku na jeho zdroj

⁷<https://docs.unity.cn/2021.3/Documentation/Manual/class-AudioListener.html>

⁸<https://docs.unity.cn/2021.3/Documentation/Manual/AudioOverview.html>

⁹<https://docs.unity.cn/2021.3/Documentation/Manual/class-AudioSource.html>

XR

Od verzie 2019.3, Unity oficiálne podporuje tieto platformy pre XR:

- ARKit
- ARCore
- Microsoft HoloLens
- Windows Mixed Reality
- Magic Leap
- Oculus
- PlayStation VR

Pre vývoj na iných platformách je možné použiť Unity XR SDK. Používajú ho napríklad spoločnosti Valve a Google pre OpenVR a Cardbord.

Výraz XR zahŕňa virtuálnu realitu (VR), ktorá simuluje úplne iné (nereálne) prostredie okolo používateľa. ďalej Zmiešanú realitu (MR), tá kombinuje nereálne (fiktívne) prostredie s prostredím reálneho sveta a umožňuje im vzájomnú interakciu. A nakoniec rozšírenú realitu (AR), ktorá na rozdiel od ostatných iba vrstvi obsah fiktívneho (digitálneho) sveta na reálny.

Pre vývoj pre platformu Oculus je možné využiť asset `Oculus Integration` na to určený. Obsahuje niekoľko predpripravených prefabov pre VR kameru `OVRCameraRig`, pre pohyb užívateľa / hráča `OVRPlayerController`, alebo napríklad implementácia vstupu z rúk `OVRHandPrefab` [21]. Multiplatformnú variantu pre vývoj predstavuje `XR Interaction Toolkit`. Ten umožňuje vyvíjať VR aplikácie pre zariadenia ako HTC Vive, Oculus, Windows, ale dokonca aj Androidové systémy.¹⁰

¹⁰<https://docs.unity3d.com/2020.1/Documentation/Manual/UpgradeGuide20201.html>

5.2 Virtuálna realita

Podľa článku od Mandal [12] (nasledujúce informácie v tejto sekcii sú prevzaté a adaptované najmä z tohoto článku), virtuálna realita (VR) je názov pre absorbujúci, interaktívny a počítačom sprostredkovaný zážitok, v ktorom osoba vníma syntetické (simulované) prostredie pomocou špeciálneho vybavenia medzi človekom a počítačom. Interaguje so simulovanými objektmi v danom prostredí, akoby boli skutočné. Niekoľko osôb sa môže navzájom vidieť a komunikovať v zdieľanom syntetickom prostredí. Virtuálna realita je termín používaný na označenie počítačom generovaného virtuálneho prostredia, ktoré môže používateľ posúvať a manipulovať s ním v reálnom čase. Virtuálne prostredie sa môže zobraziť na displeji pripojenom na hlavu, monitore počítača alebo na veľkej projekčnej obrazovke. Systémy sledovania hlavy a ruky sa používajú na to, aby používateľovi umožnili pozorovať, pohybovať sa a manipulovať s virtuálnym prostredím. Užitočnosť VR už bola skúmaná v oblasti automobilového dizajnu, dizajnu robotov, medicíny, chémie, biológie, vzdelávania, ako aj v oblasti navrhovania a konštruovania budov.

Systémy VR môžeme zoskupiť podľa úrovne ponorenia, ktorú ponúkajú používateľovi:

- Neponorujúce sa systémy (Desktop VR) - využívajú monitory (obrazovky) na zobrazenie virtuálnej reality. Sú jednoduché na použitie a nevyžadujú špeciálne zariadenia. Užívateľ môže s daným prostredím komunikovať, ale nie je v ňom ponorený.
- Čiastočne ponorné systémy (Fish Tank VR) - vylepšená verzia desktop VR. Tieto systémy podporujú sledovanie hlavy a využívajú efekt pohybovej paralaxe. Stále používajú monitor na zobrazenie prostredia, ale môžu využívať aj okuliare s LCD uzáverom pre stereoskopické sledovanie. Nepodporujú senzorický výstup.
- Ponorné systémy - umožňujú užívateľovi úplné ponorenie do sveta generovaného počítačom, ktorý podporuje stereoskopický pohľad na scénu podľa jeho polohy a orientácie. Tieto systémy bývajú často vylepšené zvukom, haptickým a senzorickým rozhraním.

Podobnými technológiami sú aj zmiešaná realita (MR - mixed reality), ktorá mieša reálny svet s virtuálnym a teda neprevláda ani jedna zložka a objekty z oboch svetov koexistujú naraz. Ďalej rozšírená realita (AR - augmented reality), ktorá rozširuje jeden svet tým druhým.

Hlavnými výhodami VR sú podľa Mandal [12] nasledujúce:

- Každý technický systém môže mať veľa funkcií, ktoré sú príliš malé alebo veľké. VR umožňuje používateľovi monitorovať, ovládať a pozorovať túto funkciu v systéme s normálnym rozsahom.
- VR umožňuje používateľovi cítiť a vnímať „nereálny čas“. Nereálny čas znamená prípad alebo situáciu ponúkanú v rýchlom alebo pomalom čase.
- Môže byť použité na zlepšenie vzdelania, výcvikových nástrojov a skúseností.
- Simuluje interakciu a jej rýchlosť alebo vyššiu rýchlosť v reálnom svete.
- Väčšina systémov vo VR dáva používateľom možnosti opakovať úlohu, kým túto úlohu nesplní profesionálne s požadovanými zručnosťami.

- Virtuálne prostredie je oveľa bezpečnejšie ako skutočné prostredie.
- Technológia VR poskytuje používateľom VR schopnosť pozorovania a monitorovania z mnohých počtov zobrazení.
- Podpora technológie VR a zdokonaľovanie dištančného vzdelávania a vyhýbanie sa skutočnému nebezpečenstvu, prekonanie časových obmedzení, poskytuje študentom množstvo vzdelávacích zdrojov, ktoré im umožňujú samostatne skúmať učenie. Zvyšuje samo-štúdium.
- VR je najviac použitá technológia na zdokonalenie inžinierskeho výcviku, navrhovania v spoločnosti a na cestách a uliciach, a to nielen v strojárstve, ale aj v medicíne a vzdelávacom prostredí, ako to vidíme v aplikáciách VR.
- VR nevyžaduje, aby používatelia prezentovali na rovnakom mieste školenia simuláciu systému, ktorú môže používateľ trénovať, aj keď je v inej krajine.
- Použitím 3D simulácie v technológii VR mohli byť kultúrne informácie prezentované v 3D modeloch z mnohých uhlov pohľadu. Umožňuje ľuďom pochopiť spoločenské vedy, krajinu a tradície skutočného sveta.

Niet však pochyb o tom, že VR čelí niektorým znateľným obmedzeniam a prekážkam (podľa článku od Alqahtani, Daghestani a Ibrahim [3]). VR umožňuje používateľom komunikovať v reálnom čase, táto výhoda vyžaduje počítačový proces na manipuláciu s virtuálnym svetom v reálnom čase. Niektoré typy systému VR vyžadujú na ich vytvorenie oveľa viac financií, napríklad v prípade systému s úplným ponorením.

VR sa využíva na zábavu (hry), vzdelávanie (lekársky alebo vojenský výcvik), vedu. V súčasnosti sa vo VR pre vytváranie realistických obrazov, zvukov a iných vnemov používajú hlavne headsety, poprípade prostredia s viacerými projekciami, ovládače a niekedy aj haptické rukavice, pohyblivé pásy či haptický oblek (obr. 5.7). Tieto nástroje simulujú fyzickú prítomnosť používateľa vo virtuálnom prostredí. Používateľ sa tak môže rozhliadnuť po simulovanom svete, pohybovať sa v ňom a interagovať s objektami v ňom.



(a) Haptické rukavice pre pocit uchopenia virtuálneho objektu



(b) Multismerový posuvný pás umožňujúci pohyb bez kontrolérov



(c) Haptický oblek simulujúci dotyk na tele vytváraný vo virtuálnej realite

Obr. 5.7: Doplnkové zariadenie ku VR setu¹¹

Pre VR zariadenia sa rozlišujú dva typy realizácie snímania: inside-out a outside-in. Inside-out zariadenia majú snímače priestoru a umiestnenia užívateľa v priestore rovno v headsete. Voči okolitému prostrediu headset zaznamenáva zmeny pomocou vstavaných kamier. Varianta outside-in vyžaduje externý hardware ako senzory snímajúce pozíciu hráča a ovládače. Tieto senzory však musia byť umiestnené veľmi špecificky podľa hardwarových parametrov daných setov. Príklad takýchto setov sú HTC Vive alebo Oculus Rift (obr. 5.8).

Hardwarové senzory headsetu následne prekladajú relatívnu polohu, smer pohľadu a jej zmeny do softwarového prostredia, ktoré využije tieto dáta pre kamerový objekt, vďaka čomu sa bude môcť vyrenderovať do headsetu práve tá časť 3D prostredia aplikácie, na ktorú sa užívateľ virtuálne pozerá. Tým je dosiahnutý pohľad prvej osoby (First Person Perspective).



Obr. 5.8: Oculus Rift (naľavo), HTC Vive (napravo)¹²

Kontroléry pre VR set sú bežne dva, pre každú ruku jeden. Majú na sebe rôzny počet tlačidiel v závislosti od výrobcu setu, niektoré majú aj joysticky pre senzitívnejšie ovládanie niektorých akcií. VR kontroléry sú určené na interakciu s virtuálnym prostredím, či už na manipuláciu s objektami, prostredím alebo špeciálnejšie prípady ako napríklad pohyb po VR prostredí. V softwarovej reprezentácii v 3D VR aplikácii najbežnejšie možno vidieť kontroléry ako ruky užívateľa.

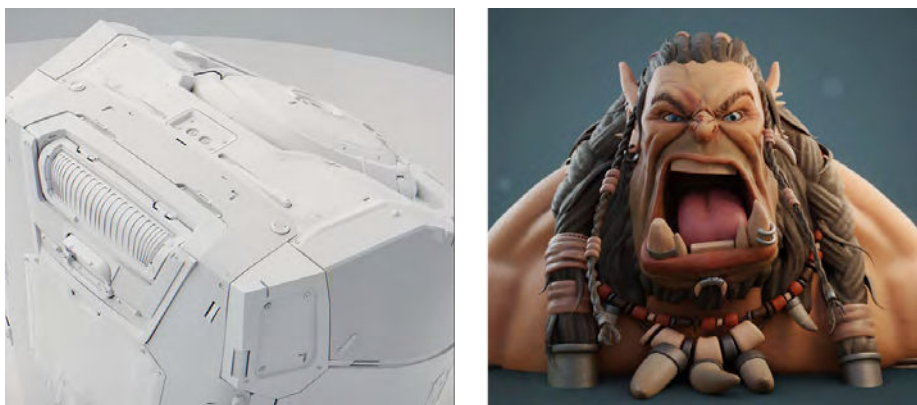
¹¹<https://teslasuit.io/blog/vr-glove-by-teslasuit/>, <https://invest.virtuix.com/>, <https://teslasuit.io/the-suit/>

¹²<https://store.hp.com/us/en/cv/omenvrpcs>

5.3 Blender

Blender je bezplatná a open source aplikácia pre tvorbu 3D. Podľa internetového zdroja od *The Blender Foundation* [6], táto aplikácia podporuje celú škálu 3D pipeline - modelovanie, rigging, animáciu, simuláciu, vykresľovanie, skladanie a sledovanie pohybu, dokonca aj strih videa a tvorbu hier. Pokročilí používatelia používajú Blender API pre skriptovanie v Pythone na prispôsobenie aplikácie a tvorbu špecializovaných nástrojov. Blender je vhodný pre jednotlivcov a malé štúdiá, ktoré profitujú z jeho zjednoteného procesu a procesu responzívneho vývoja. Blender je multiplatformový a funguje rovnako dobre na počítačoch Linux, Windows a Macintosh. Jeho rozhranie používa OpenGL na zabezpečenie konzistentného zážitku. Ako projekt riadený komunitou v rámci GNU General Public License (GPL) je verejnosť oprávnená vykonávať malé aj veľké zmeny v kóde programu, čo vedie k novým funkciám, opravám chýb a lepšej použiteľnosti.

Pre vytvorenie objektu sa používajú väčšinou dve metódy a to je modelovanie a sculpting (sochárstvo) (paragraf adaptovaný z knihy *Blender master class: A hands-on guide to modeling, sculpting, materials, and rendering* [15]). Obe metódy sú využiteľné samostatne no často je využitá kombinácia oboch. Modelovanie je vhodné hlavne na vytváranie objektov s tvrdým povrchom, vytváranie hrán, využitie paralelných čiar a uhlov. Môže tak vytvoriť veľmi presné modely (obr. 5.9, vľavo). Sculpting na druhú stranu je vhodnejší na jemné, oblé hrany, mäkké a organické objekty. Spojením týchto metód je možné vytvoriť veľmi realistické objekty (obr. 5.9, vpravo).



Obr. 5.9: Modelovanie objektu s tvrdým povrchom (vľavo)¹³, sculpting organického objektu (vpravo)¹⁴

Pri modelovaní pomocou meshov sa väčšinou začína s primitívnymi tvarmi (kruh, guľa, kocka, cylinder, ...), zložených z polygónov. Vďaka tomu je možné upraviť každý vrchol, hranu alebo plochu a tak vytvárať zložitejší objekt. Úpravu meshu je teda možné robiť pomocou základných funkcií pre ich komponenty alebo použitím modifierov na úpravu, generovanie, deformáciu a simuláciu. Je možné taktiež pracovať s krivkami, povrchmi (ako rozšírenie kriviek v 3D), metaballmi, mriežkami a textovými objektami, podľa typu objektu, na ktorý sú tieto metódy vhodné. Sculpting využíva na úpravu meshu hlavne štetce, tak ako pri farbení, ale na rozdiel od zmeny farby, upravuje vrcholy, hrany a plochy v rozsahu štetca. Vďaka tomu je úprava rýchlejšia než presúvanie jednotlivých vrcholov. Nevýhodou

¹³<https://blendermarket.com/products/hard-surface-modeling-in-blender>

¹⁴<https://cgcookie.com/u/sebastiancavazzoli/projects/durotan-sculpt-in-blender>

môže byť však nedostatok presnosti, potreba tvoriť ostré rohy a hrany a úprava menších delikátnych oblastí, pri ktorých by mohli byť deformované aj nesprávne časti. Pre zamedzenie deformácii niektorých oblastí, je možné tieto oblasti označiť maskou alebo celé skryť (paragraf adaptovaný z tutoriálu od Taylor [16]).

Ďalšiu dôležitou časťou sú materiály, textúry a procedurálne generované shadery. Materiály popisujú vzhľad objektu, definujú farbu, textúru a látku, z ktorej je predmet vyrobený a ako s ním interaguje svetlo. Fyzikálne založené materiály je možné vytvárať pomocou shaderov. Sú vhodné na vytváranie plastu, skla, kovu, látky, kože, vlasov, dymu a ohňa. Materiály je možné vytvoriť buď pomocou nastavenia vlastností vstavaných shaderov alebo v editore shaderov. Textúry sú väčšinou bitmapové obrázky, ktoré sa používajú na vyfarbenie povrchu meshu. Môžu byť vytvorené buď v Blendery alebo aj v inej externej aplikácii. Tieto textúry sa potom mapujú na povrch objektov, alebo sú využité v zložitejších materiáloch a shaderoch (paragraf adaptovaný z manuálu od spoločnosti *The Blender Foundation* [17]).

Vytvorené objekty je možné uložiť ako blender súbor alebo exportovať v jednom z podporovaných formátov (napríklad fbx alebo obj). Tieto súbory je potom možné použiť v iných programoch, akým je napríklad aj Unity 3D.

Kapitola 6

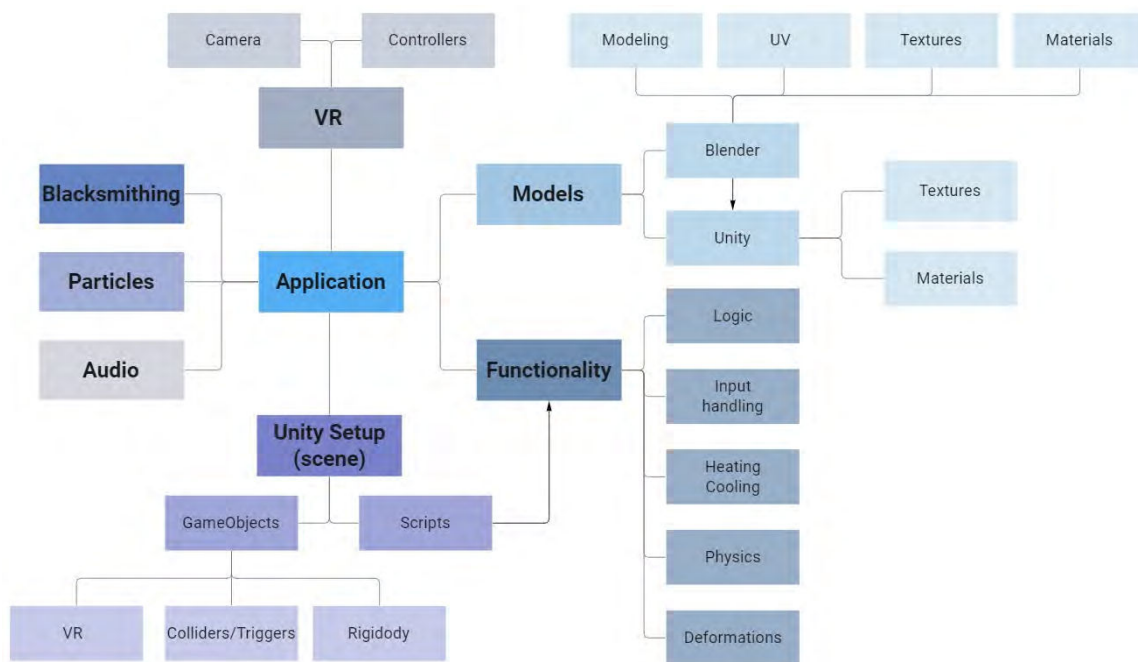
Návrh aplikácie

Táto kapitola obsahuje návrh aplikácie umožňujúcej virtuálnu demonštráciu kováčskych techník a jej jednotlivých častí. Taktiež upresňuje cieľ aplikácie. Popísané sú základné návrhy, ktoré demonštrujú základnú myšlienku aplikácie, ale aj postupný rast a úpravu návrhov na základe získaných dát a informácií.

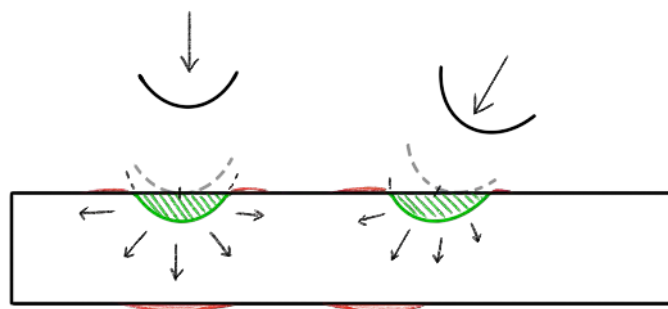
6.1 Koncept

Kováčstvo vo VR je pomerne komplexná téma a aplikácia ako taká zahŕňa množstvo častí, ktoré sú navzájom prepojené. Diagram 6.1 zobrazuje tieto časti ako aplikačné bloky a ich podčasti, ktoré sú na sebe rôzne závislé na rozličných úrovniach. Hlavné časti, ktorými sa návrh zaoberá, sú kováčstvo, scéna, modely, virtuálna realita, výsledná funkcionálna a doplnkové efekty.

Základnou témou aplikácie je kováčstvo, teda pri navrhovaní je nutné zohľadniť, akými kováčskymi technikami sa možno inšpirovať, a aké kováčske nástroje by mohli byť v aplikácii využité. Metódy a techniky na virtuálne kováčstvo boli vybrané podľa podobného typu deformácie z hľadiska programovania. Pri návrhu je kladený dôraz na schopnosť aplikácie zreplikovať kováčstvo ako voľnú deformáciu kovového objektu bez predom pripravených stavov daného kovu. Tým má užívateľ plnú kontrolu nad virtuálnym kutím – z toho dôvodu však vzniká množstvo potenciálnych problémov a situácií, ktoré treba potom zohľadniť a nájsť vhodné riešenie.



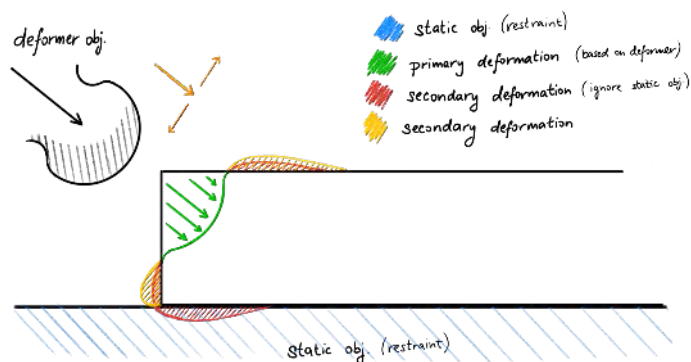
Obr. 6.1: Rozdelenie aplikácie na 7 hlavných častí: Kováčstvo, VR, Modely, Unity Setup, Funkcionalita, Častice, Audio. Kováčstvo predstavuje rôzne nástroje a typy/postupy deformácie kovového telesa. Virtuálna realita je rozdelená na vizuálnu (kamera) a interaktívnu časť (kontroléry na ovládanie aplikácie). Modely v scéne sú modelované v programe Blender, vytvorené pre ne materiály a nastavené UV pre správnu vizualizáciu materiálov. Tieto modely sú použité v engine Unity, materiály sú upravené alebo vytvorené pre potreby tohto systému. Objektom v scéne sú nastavené komponenty na základe typu a funkčnosti ako kolízne telesá, ovládanie pomocou VR poprípadne nastavenie fyzikálnej odozvy, alebo vlastné definované skripty. Tie umožňujú aplikácii fungovať, ovládať jednotlivé objekty, riešiť kolízie a deformácie. Vďaka týmto vlastným a preddefinovaným skriptom je zrealizovaná celková funkcionálna aplikácia. Vstavané skripty zaisťujú fungovanie základných prvkov ako je spracovanie vstupu a fyzikálny systém v priestore. Vlastné skripty zrealizujú hlavnú logiku aplikácie, ako napríklad deformácie, nahrievanie a ochladzovanie kovu.



Obr. 6.2: Ilustrácia pôsobenia sily úderu na základe jeho smeru

Deformácia telesa prebieha v reálnom čase a teda dôležitú rolu hrá úroveň detailu 3D modelov kovových objektov. Tiež je kľúčové, aby sa kovový objekt deformoval presne v bode kolízie s kladivom a so zodpovedajúcou aplikovanou silou (obr. 6.2) s ohľadom na vytvorenie

čo najrealistickejšieho deformovania kovu (obr. 6.3). Okrem toho sú riešené rôzne limitácie ohľadom deformovania, a teda spracovávania kolízií. Z dôvodu spôsobu implementácie knižnice pre VR kontroléry vzniká problém ohľadom kutia železa pri jeho držaní rukou (ovládačom), pretože uchopením predmetu sa zmenia fyzikálne vlastnosti daného objektu. Pri zaobchádzaní môže užívateľ vytvoriť rôzne situácie, pri ktorých by nemalo byť možné kovať kovový objekt, ako napríklad počas držania kovu prekryvajúceho sa čiastočne s nákovou, na ktorej je kovanie ako jediné miesto povolené. V súvislosti s kolíziami je tiež nutnosť vhodne nastaviť kolízne telesá jednotlivým objektom vystupujúcim vo virtuálnej kovárni. Niektoré kombinácie kolízií medzi telesami nemusia byť zachytené systémom (kolízne telesá v tvare konkrétneho meshu daného objektu), a teda sa kladie dôraz na efektívne a vhodné zvolenie týchto kolíznych telies. Ďalší problém predstavuje kombinovanie uchopiteľného predmetu s kinematickým, ktorý má za úlohu vytvárať kolízie – kladivo, pozostávajúce z rúčky a hlavy.



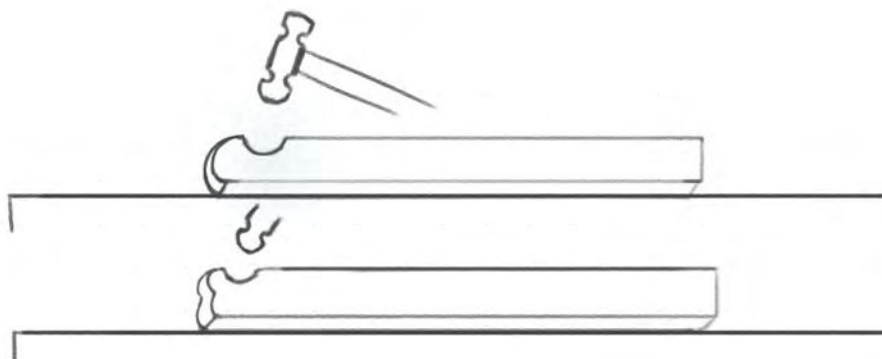
Obr. 6.3: Konceptuálny návrh realizácie deformovania kovového materiálu s ohľadom na nákovu

Pre kováčske predmety je potrebné správne navrhnuť ich vizuálnu reprezentáciu, funkcionality a spôsob vloženia do systému, v ktorej sa aplikácia vyvíja. Okrem kováčskych nástrojov je predmetom návrhu aj vytvorenie scény, ktorá má za úlohu imitovať kováčsku dielňu, a teda má obsahovať príslušné zariadenia. Jednotlivé objekty majú svoju jednoznačnú úlohu a teda je im potrebné nastaviť správne chovanie voči ostatným objektom v scéne. Z hľadiska realistickej vizualizácie predstavuje priradovanie materiálov k 3D modelom, hlavne ak chceme dosiahnuť vzhľad dynamickej textúry na istých objektoch, ako sú voda v nádobe a nahriate kovové teleso. Aby bolo možné kov nahrievať realisticky, teda iba tie časti, ktoré sú skutočne vložené do ohňa, možno nahradiť textúrou procedurálnym materiálom, ktorý je však potrebné správne prepojiť s logikou aplikácie. Jedna z posledných dôležitých oblastí, ktorú je potrebné riešiť, je rozšírenie užívateľského zážitku pomocou ďalších vnemov, ktoré ponúka technológia VR. Jedná sa o vizuálne prvky – okrem procedurálnych materiálov využitie časticových systémov, zvukové prvky a hmatové odozvy pri práci s virtuálnym kovaním.

6.2 Kováčstvo

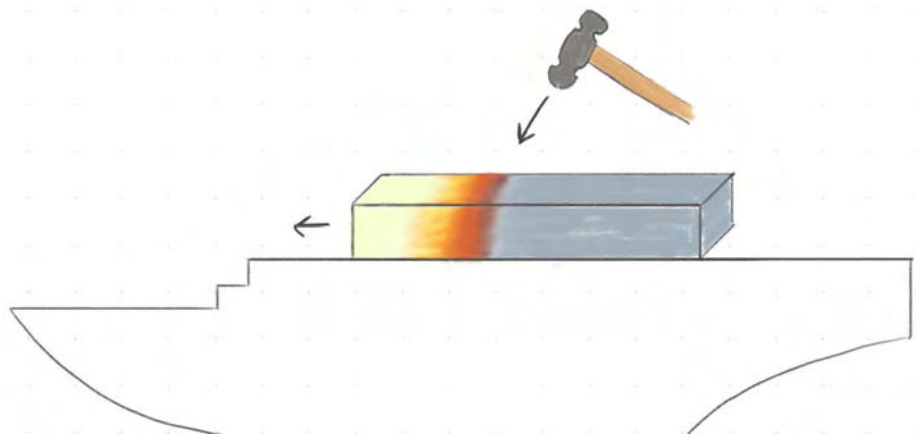
Kováčstvo zahŕňa rôzne typy deformácií, pri ktorých závisí na viacerých premenných. Z programátorského pohľadu je možné prerozdeliť tieto premenné (závislosti) nasledovne:

- typ kladiva
 - hmotnosť a veľkosť kladiva
 - tvar hlavy kladiva
- tvar a veľkosť kovového materiálu
- úroveň tepelného zahriatia materiálu (v rádoch stoviek stupňov Celzia)
- umiestnenie, sila a uhol dopadu kladiva na materiál (obr. 6.4)
- umiestnenie materiálu na nákov
 - pozícia polohy materiálu
 - veľkosť plochy dotyku materiálu o nákovu
 - uhol materiálu voči nákovu
 - umiestnenie zahriatej časti medzi nákovou a kladivom



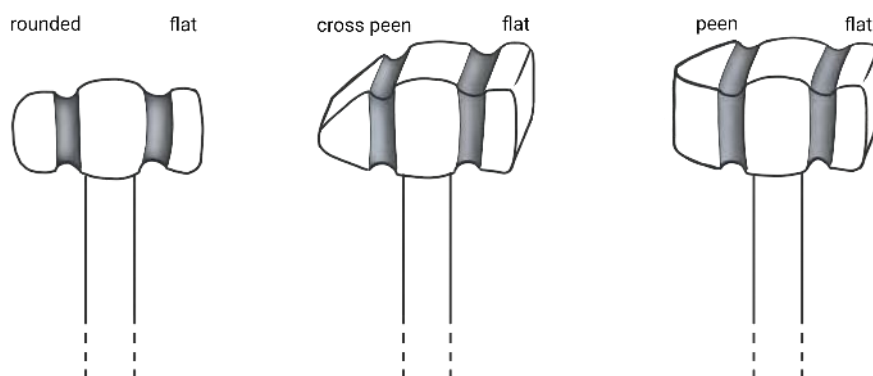
Obr. 6.4: Ukážky deformovania kovového materiálu na základe miesta dopadu kladiva na kov. Vrchný obrázok zobrazuje vertikálny dopad kladiva, spodný obrázok zobrazuje šikmý dopad kladiva.

Stále však závisí na tom čo sa kováč rozhodne robiť, a akú techniku použije. Základné kováčske techniky boli popísané v kapitole 4 o kováčstve. Tieto techniky sú často veľmi rozdielne z pohľadu práce s fyzikálnym objektom a jeho naprogramovania. Niektoré napríklad vyžadujú rozdelenie deformovaného kovového objektu (teda jeho meshu) na viaceré časti, niektoré zas ohýbanie alebo rotáciu danej časti, alebo deformáciu objektu pomocou úderov. Pre každú je potrebné navrhnuť iný typ algoritmu, ktorý by deformovanie objektu na základe techniky dokázal implementovať. Ak sa zameriame na všeobecné základy so zredukovaným počtom potrebných nástrojov alebo použitia špecifických nástrojov a ich rozšírení, dostaneme sa k deformácii kovového objektu s kladivom na nákovu (obr. 6.5).



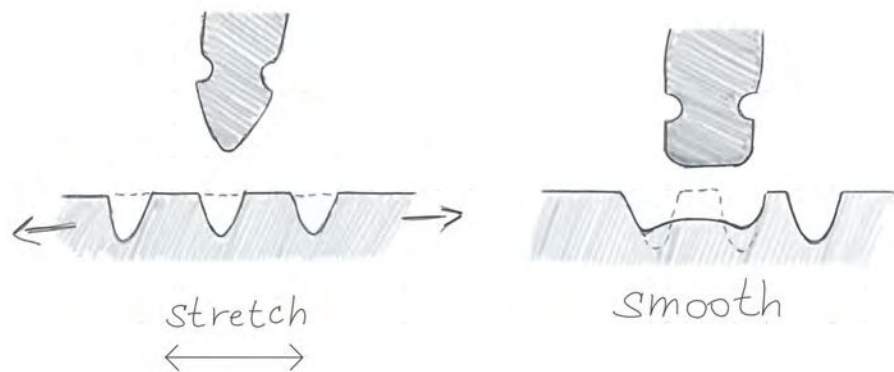
Obr. 6.5: V okolí bodu úderu kladiva na kovový materiál sa má kov deformovať a zároveň rozťahovať do priestoru a smeru s najmenším odporom

Na základe tvaru hlavy (označovaného odborné tiež ako nosu) kladiva závisí, do akých smerov sa kov deformuje a s akou veľkou silou, poprípade ktorými kladivami sa dá najlepšie dosiahnuť cielený výsledok. Do úvahy sú brané najmä kladivá určené na deformáciu kovu, nie jeho prebíjanie alebo sekanie. Kladivá, ktoré sa často používajú na základe tvaru (nie veľkosti) sú zaoblené (rounded), ploché (flat), kladivá s rovnobežným nosom (peen) a s priečnym nosom (cross peen) a oblé (ball peen) (obr. 6.6).



Obr. 6.6: Zľava kladivo s oblou a plochou hlavou, kladivo s priečnym nosom a plochou, kladivo s rovnobežným nosom a plochou

Čím je plocha hrotu hlavy menšia, tým výraznejšie vytvára deformácie a naopak čím je väčšia a plochšia tým sú deformácie menšie. Plochá hlava sa používa najmä na vyrovnávanie väčších deformácií, ktoré spôsobili viac zaoblené kladivá (obr. 6.7).



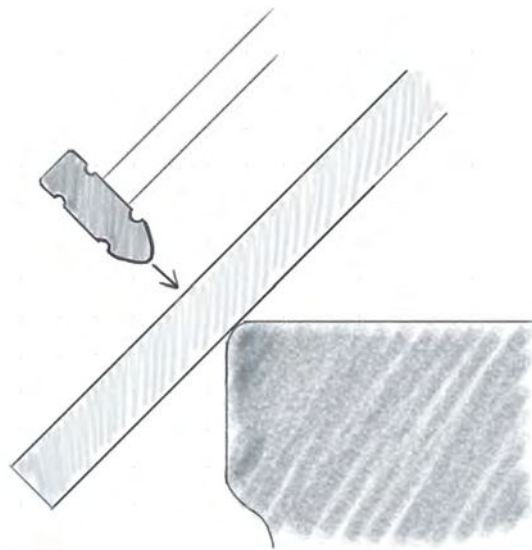
Obr. 6.7: Vľavo väčšie deformácie s použitím kladiva s väčším zaoblením a menším povrchom dotyku, využívané na rozťahovanie. Vpravo vyhladzovanie povrchu kovu za použitia plochej hlavy kladiva.



Obr. 6.8: Vľavo detail na hlavu plochého kladiva určeného na vyhladzovanie povrchu, vpravo porovnanie starého a nového moderného kladiva¹

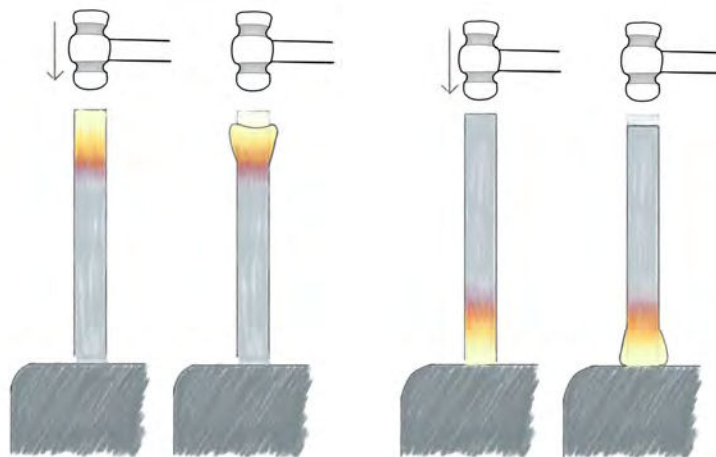
Ako bolo vysvetlené v kapitole 4, Kováčstvo (časť Nástroje), nákovy je používaná na deformáciu kovu spolu s kladivami. Slúži ako podložka, ale súčasne na kov pôsobí zdola alebo zo strán ako kladivo, podľa toho, kde sa na nej udiera (obr. 6.9, kov priložený v 45° ku povrchu nákovy na jej hrane, kladivo udiera kolmo na povrch kovu zrkadlovo v mieste priloženia kovu ku nákovy).

¹<https://alecsteeleshop.com/collections/hand-tools/products/the-square-circle-rounding-hammer>



Obr. 6.9: Ukážka pred deformáciou na hrane nákovy

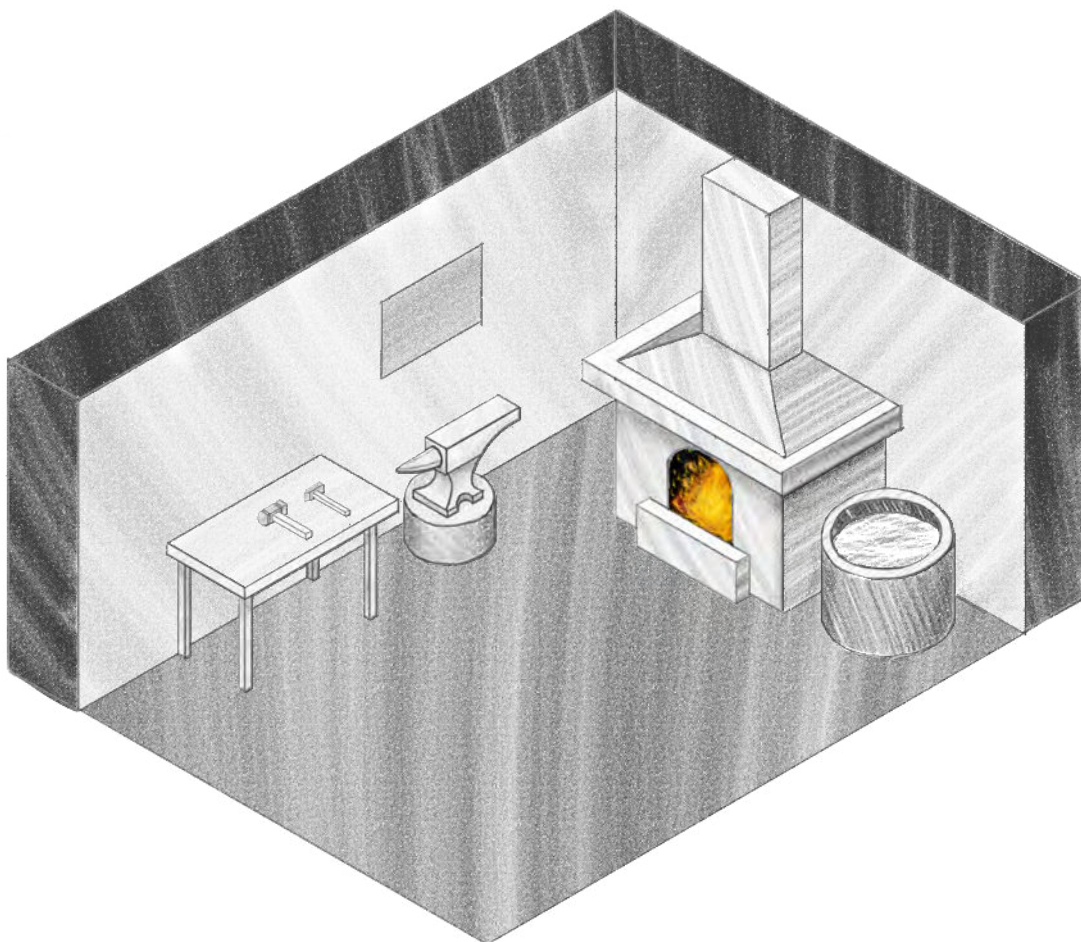
Ďalšia dôležitá časť je tepelné nahrievanie kovu, bez ktorého je s ľudskou silou takmer nemožné dosiahnuť výraznejšie deformácie materiálu. S nahriatím kovu sa teda počíta pri každej deformácii. Nahriatie ovplyvňuje nie len samotnú silu deformácie ale aj plochu pôsobenia. Tiež záleží, kde je nahriata časť umiestnená (obr. 6.10).



Obr. 6.10: Z ľava pred a po deformácii nahriatej časti nedotýkajúcej sa nákovy, vpravo pred a po deformácii kovu, keď sa nahriata časť dotýka nákovy

6.3 Scéna, Modely, Materiály

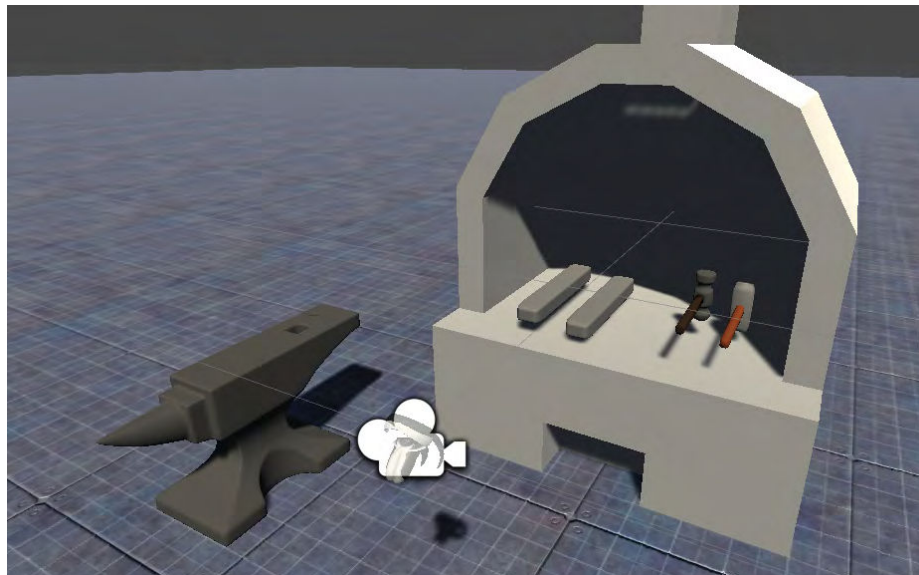
Pri tvorbe VR aplikácie je zásadné vziať do úvahy, že sa jedná o 3D grafickú aplikáciu, ktorá pozostáva zo scény a modelov. Pre vytvorenie 3D priestoru kováčskej dielne je potrebné myslieť na to, že aplikácia je vo VR a teda scéna by nemala zaberať príliš veľký priestor. Vo VR sa užívateľ môže premiestňovať rôznymi spôsobmi (viac popísané v sekcii 6.4 Virtuálna realita). Ak by bola scéna veľmi veľká, užívateľ by sa musel pohybovať príliš veľa po priestore aplikácie, a z hľadiska požiadaviek takejto virtuálnej kováčskej dielne to môže byť zbytočné. Rovnako v reálnej kováčskej dielni je rozumné kompaktné usporiadanie pracovných plôch, aby materiál pri prenose zbytočne nechladol.



Obr. 6.11: Ilustrácia návrhu scény, zľava stôl s kladivami, nákova, vyhňa alebo pec a sud s vodou

Návrh scény

Scéna má obsahovať základné objekty potrebné pre vizuálny a herný zážitok. Medzi tieto objekty patria predovšetkým miestnosť, pec, nádoba s vodou, nákova, kladivá a kovové materiály (obr. 6.11). Tie sú nevyhnutné lebo všetky sú využívané pri deformácii kovu.



Obr. 6.12: Základná scéna

Na prvotnom návrhu scény (obr. 6.12) je umiestnená pec a nákova ako základné statické objekty. Najdôležitejším nástrojom a objektom v aplikácii je kladivo určené na deformáciu kovového materiálu. V kovárskej dielni sa väčšinou vyskytujú viaceré kladivá, podľa potreby rôznych typov deformácii kovu. Poslednou generickou súčasťou scény je samotný kovový materiál, ktorý má počiatkový tvar z ktorého sa vychádza.

3D modely objektov

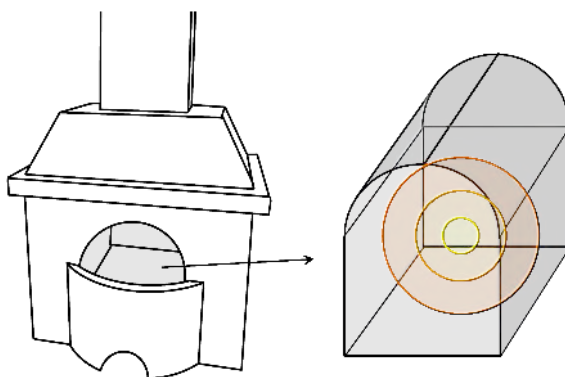
Pre jednotlivé objekty v 3D aplikácii je potrebné vymodelovať ich štruktúru (sieť, mesh), vytvoriť a nastaviť materiál tak, aby výzor týchto objektov bol čo najrealistickejší. Pre vytváranie modelov je možné použiť rôzne techniky určené či už pre objekty s tvrdým povrchom alebo skôr organické objekty. Väčšina objektov v scéne spadá pod typ objektov s tvrdým povrchom. Jediný objekt, ktorý môže byť považovaný skôr za organický, je kov, ktorý má mať schopnosť byť deformovaný.

Materiály pre modely je možné vytvoriť pomocou jednoduchých farieb, využitím textúr alebo vytvoriť procedurálne spojením logických operácií a daných hodnôt. Je možné ich taktiež spojiť a použiť naraz. Pre vytvorenie realistickejšej vizualizácie je vhodné použiť buď textúry z reálnych objektov alebo ich vytvoriť procedurálne.

Pec

Pec má za úlohu nahriať kovový materiál na dostatočne vysokú teplotu tak, aby ho bolo možné deformovať. Na tento účel je nutné aby bolo možné do pece vkladať predmety a k tomu prispôbiť kolízne telesá. Pre uskutočnenie nahrievania kovu sa musí zisťovať, či je daný kov vložený do pece. Na tento prípad sa dá využiť kolízne teleso nastavené ako trigger,

pretože nepotrebuje zisťovať fyzikálne hodnoty, iba či je do neho vložený objekt. Keď je kov vložený do pece, vypočítava sa vzdialenosť jednotlivých častí kovu k epicentru ohňa (obr. 6.13). Časti kovu, ktoré sú bližšie ku stredu ohňa sa nahrieva rýchlejšie než pri jeho okraji. Pec je v scéne nepohyblivé statické teleso, teda sa s ním nejako nemanipuluje a tiež nemá byť pre užívateľa možné, aby cez ňu prešiel. Pec je navrhnutá tak, aby bolo možné položiť na ňu rôzne nástroje a kovový materiál, teda čiastočne slúži ako odkladací priestor. Ďalšia možnosť rozšírenia pece je pridať mech, ktorým sa rozducháva oheň, prípadne rozšírenie o miesto určené na uhlie alebo vodu na regulovanie sily ohňa a teda dodávania teploty kovovým materiálom.

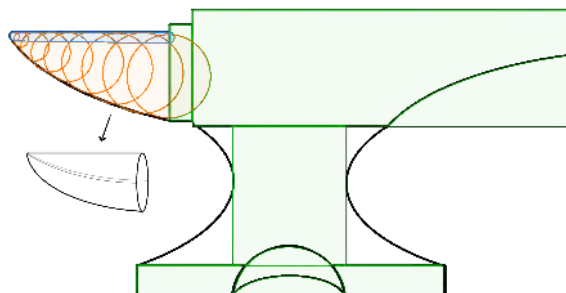


Obr. 6.13: Pec a detail na vnútorný priestor pre nahrievanie kovov. Žltý kruh je stred ohňa s najvyššou teplotou a najrýchlejším nahrievaním, oranžový vonkajší kruh je na okraji ohňa a nahrieva kov pomalšie.

Nákova

Veľmi dôležitý statický predmet vo virtuálnej kovárskej dielni je nákova, ktorá slúži ako podložka pre umiestnenie kovu. Na nákovu sa deformuje nahriaty kovový materiál, využíva sa aj ako deformačné teleso z opačnej strany než je účinok kladiva. Pri nákovu je dôležité, aby mala vrchnú časť čo najviac rovnú, s jemne zaoblenými hranami. Ďalej, takzvaný stôl, ktorý vyzerá ako schod oproti hlavnej ploche. Obsahuje jeden alebo viac oblých rohov podľa typu nákovy. V plochej časti sú taktiež umiestnené diery: jedna väčšia štvorcová, určená pre rôzne pomocné nástroje alebo ako pomocná diera na prebíjanie, druhá diera je menšia, kruhového tvaru, a je určená pre prichytávacie pomocné nástroje.

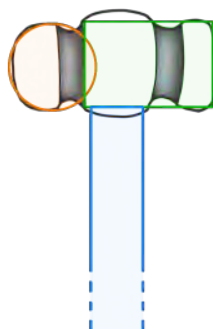
Kvôli komplikovanému tvaru by bolo vhodné použiť kolízne teleso zhodné s meshom nákovy. V Unity frameworku však nie je možné zisťovať kolízie medzi dvoma kolíznymi telesami tohto druhu. Takýto druh sa používa už pri objektoch kovových materiálov, z toho dôvodu sú kolízne telesá pre nákovu zjednodušené nahradením za kombináciu základných tvarov. Na nákovu sú 3 najdôležitejšie časti pre deformáciu: hlavná plocha, schod a roh (obr. 6.14). Hlavná plocha a schod majú tvar kváдру v oblasti kde je kov deformovaný, a tak je možné využiť boxové kolízne telesá. Pre roh, ktorý je kužeľovitého tvaru s naklonenou osou sú využité guľové kolízne telesá a kolízne teleso kapsulového tvaru. Kapsulové teleso je použité na vrchnej časti rohu, aby bol povrch rovný a nezanechával zuby. To by sa dalo vyriešiť aj viacerými kolíznymi telesami guľového tvaru, ale to by výrazne ovplyvnilo čas detekcií kolízií. Zvyšné časti nákovy už nie sú používané na deformáciu a preto sú využité jednoduché boxové kolízne telesá kvádrového tvaru na podstavu.



Obr. 6.14: Nákova a jej zjednodušená reprezentácia pomocou kolíznych telies. Zelené - kvádrové teleso, oranžové - guľové teleso, modré - kapsulové teleso.

Kladivo

Kladivo sa používa ako deformačný objekt, ktorý je zložený z dvoch častí – z kovovej hlavy a drevenej rúčky. Hlava kladiva je určená na deformáciu a tak obsahuje kolízne telesá ktoré kopírujú tvar hlavy kladiva čo najvhodnejšie. Podobne ako pri nákovе nie je možné použiť kolízne teleso z meshu, preto sú využité jednoduché telesá pre popis (obr. 6.15). Pre oblú časť kladiva je vhodné guľové kolízne teleso a pre zvyšnú časť hlavy boxové kolízne teleso. Deformáciu kovového objektu vykonávajú len kolízne telesá hlavy kladiva. Rúčka kladiva je určená na jeho uchopenie, teda nepotrebuje komplikované kolízne teleso a tak je použité teleso kvádrového tvaru. Aby bolo možné kovať obidvoma stranami kladiva, je možné chytiť rúčku na ľubovoľnom mieste pod ľubovoľným uhlom.



Obr. 6.15: Kladivo a kolízne telesá na ňom použité. Zelené - kvádrové teleso, oranžové - guľové teleso a modré - kvádrové teleso odlíšené pretože sa naň funkčne pozerá ako na druhý objekt.

Kov

Objekt kovového materiálu predstavuje hlavný dynamický predmet. Jeho najdôležitejšou vlastnosťou je schopnosť byť deformovaný kladivom. 3D model kovu môže byť vytvorený s rôznymi tvarmi a rôznou úrovňou detailu. Pod úrovňou detailu sa rozumie hustota meshu, teda počet vertexov obsiahnutých v modeli. Pri vyššej hustote narastá jemnosť výsledných deformácií, avšak pri prílišnej hustote sa zvyšuje výpočtová náročnosť pri každom údere. Tento objekt ako jediný musí použiť kolízne teleso v tvare aktuálneho meshu kovového materiálu, inak by nebolo možné správne zisťovať kolízne body spôsobené kladivom. Spôsobená deformácia je rozdelená na dva druhy – primárnu, ktorá vytvorí priehlbnu v bode

dotyku kladiva s kovom, a sekundárnu, ktorá rozpína kovový materiál do ostatných strán. Pre umožnenie deformovania objektu musí byť kov umiestnený na nákove.

Na deformáciu je potrebné kov nahriať. Od teploty kovu závisí sila deformácie. Materiál tohto kovového objektu je tiež vytvorený špeciálne ako procedurálny, vďaka čomu nahriate časti kovu vyzerajú rozpálené. Kovový objekt nachádzajúci sa mimo pece sa prirodzene pomaly ochladzuje, čím sa znižuje efektívnosť kútia materiálu. Pre okamžité schladenie užívateľ vloží kovový objekt do nádoby s vodou.

Nádoba s vodou

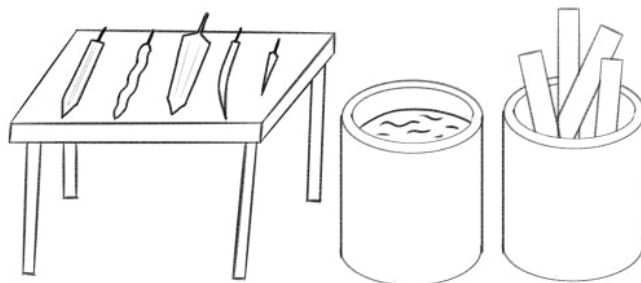
Nádoba s vodou je určená na schladenie nahriateho kovu (obr. 6.16, v strede). V rámci scény by mala mať takú veľkosť, aby sa do nej zmestil kovový materiál bez väčších problémov, ale nezaberala príliš veľa priestoru. Keďže sa jedná o ďalší statický objekt, musí mať nastavené kolízne teleso pre zabránenie prechodu užívateľa cez tento objekt.

Keďže nádoba je konkávneho tvaru a presahuje hladinu vody, nie je možné použiť kolízne teleso kopírujúce mesh objektu. Riešením je využiť viacero kolíznych telies doskového tvaru umiestnených dokola a zo spodnej strany.

Ďalšou možnosťou by bolo použiť jednoduché boxové kolízne teleso, čo by však znemožnilo vloženie kovu do vody, teda schladenie by muselo prebehnúť len za držania kovu pri vložení do vody cez kolízne teleso. Pre vodu tiež nesmie byť kolízne teleso typu mesh (aj keď je konvexného tvaru), pretože kolíziu medzi dvoma mesh kolíznymi telesami Unity nezaznamenáva. Z druhov jednoduchých kolíznych telies je najvhodnejší kapsulový tvar, keďže valcový typ neexistuje. Voda v nádobe môže byť vizualizovaná pomocou procedurálneho materiálu pre realistickjšie zobrazenie.

Pomocné objekty

Ďalšie objekty môžu rozšíriť funkčnosť alebo vylepšiť zážitok z používania aplikácie pre užívateľa. Príkladom je používanie nákovy – realistický model sám o sebe nemá veľké rozmery, avšak ak by bola nákova položená v scéne na zemi, užívateľ by sa musel pri virtuálnom kovaní zohýbať. Tiež by nedávalo veľmi realistický dojem nechať kovadlinu voľne visieť v priestore. Rovnakým spôsobom je nutné zamyslieť sa nad rozložením dostupných nástrojov pre používanie aplikácie (kovy, kladivá). Vďaka pomocným objektom ako sú napríklad stoly alebo police môžeme vyriešiť umiestnenie nástrojov (obr. 6.16), a pre nákovu môžeme použiť napríklad vymodelovaný kus dreveného pňa.

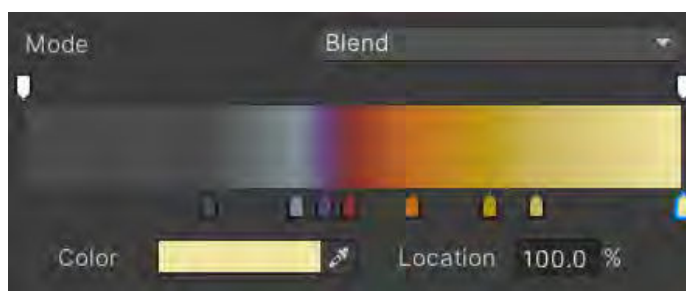


Obr. 6.16: Stôl na ukladanie výrobkov alebo nástrojov, nádoba s vodou na schladenie a nádoba na kovové objekty

Vizualizácia 3D objektov

Objekty, ktorým sa mení vzhľad za behu aplikácie potrebujú aj odpovedajúci materiál. Tento materiál je teda potrebné upravovať a prispôbovať. Ďalším dôvodom môže byť nutnosť ovplyvňovať vizualizáciu materiálu na základe stanovených parametrov. V tomto prípade nie je vhodné použiť iba 2D textúry pre vytvorenie materiálu, pretože by takýto spôsob vyžadoval veľké množstvo variácií týchto textúr. Oveľa praktickejšie je buď vytvoriť procedurálny materiál čisto pomocou logických operácií alebo aj so spojením s textúrami. Vytvoriť realistický procedurálny materiál je však celkom náročné, ak majú byť imitované reálne organické textúry. Pre vytváranie shaderov a z nich materiály je možné použiť rozšírenie do Unity zvané *Shader Graph*. Vďaka nemu sa dajú jednoduchšie vytvárať shadery pomocou vizuálnych grafov. Tieto grafy pozostávajú z funkčných uzlov a parametrov, ktoré sú poprepájané a takýmto spôsobom je možné vytvoriť veľa rôznych statických alebo dynamických materiálov.

Vo frameworku Unity sa dá vytvoriť procedurálny materiál pomocou funkcie *Shader Graph*. Pomocou kombinácie rôznych parametrov a uzlov je možné vytvoriť rôzne statické alebo dynamické materiály. Materiál pre nahrievanie kovu má mať možnosť zmeniť farbu (na základe gradientu nahrievania na obrázku 6.17) v oblasti zahrievania kovu, a teda signalizácie miesta, ktoré je možné jednoduchšie deformovať.



Obr. 6.17: Gradient vytvorený pre nahrievanie kovového objektu

6.4 Interakcia a navigácia

Dôležitou časťou v návrhu scény je pohyb v scéne a interakcia s objektami. Keďže je to hra pre VR, je nutné vybrať správny spôsob pohybu a zabrániť tak nevoľnosti z používania VR, takzvanej *VR sickness*. Podobne to platí pre interakciu s objektami. Keďže sa jedná o aplikáciu, ktorá by sa mala čo najviac blížiť realite, všetky teleportácie a otáčanie o 90° v scéne sú nežiaduce.

Frameworky a enginy pre VR

Ako už boli spomenuté softwarové reprezentácie daného VR setu, pre sfunkčnenie a zasadenie VR hardwaru do prostredia 3D grafickej aplikácie sú potrebné rôzne softwarové balíky a knižnice. Takýchto implementácií pre pridanie VR komponent do existujúceho softwarového frameworku pre 3D je pomerne veľké množstvo. Sety ako také poskytujú frameworky, knižnice alebo iné druhy softwarového rozhrania pre ich naprogramovanie do aplikácii využívajúce VR. Zároveň však existujú už aj knižnice vytvárajúce generickejšiu implementáciu VR rozhrania, ktoré je takto funkčné pre viac typov setov. Vďaka tomu možno hneď využiť aplikačné rozhranie takýchto knižníc do 3D frameworkov a sústrediť sa tak viac na samotné vytvorenie aplikácie.

Ako najznámejšie enginy/frameworky pre tvorbu 3D aplikácii sa ponúkajú 2 možnosti: Unity 3D a Unreal Engine. Obidva tieto frameworky majú v podstate veľmi podobné funkcie a prípady použitia. Pre aplikáciu tvorenú v tejto práci je dôležitá podpora pre virtuálnu realitu. V súčasnom stave túto možnosť podporujú obidva frameworky. Ďalšou dôležitou otázkou je podpora deformácie meshu 3D objektu. Vhodnejšie by bolo využiť knižnice OpenGL, avšak z dôvodu vývoju 3D VR aplikácie sa nedá brať do úvahy ako možnosť.

Unity framework umožňuje vyvíjať aplikácie pre rôzne VR platformy. Ako ďalší krok je v rámci návrhu potrebné zohľadniť dostupné technológie VR setov, ktoré potom sú pre fungovanie aplikácie kľúčové. V dnešnej dobe je týchto zariadení pomerne veľké množstvo a aj rôznych typov. Výhodou sú samozrejme zariadenia, ktoré nepotrebujú veľa prípravy a nie sú viazané na konkrétnu miestnosť, teda inside-out varianta snímania headsetu. Pre vývoj aplikácií pre VR v Unity je tiež potrebné pridať balík (*asset*) umožňujúci zobrazenie a interakciu s aplikáciou, pretože sa v ňom natívne nenachádza. Počas tvorby aplikácie sú vyskúšané dva balíky s touto funkcionalitou, *Oculus Integration* a *XR Interaction Toolkit*. Počas návrhu a vývoja aplikácie bol využívaný *Oculus Integration*, ale počas prípravy vyšiel novší balík *XR Toolkit*, ktorý umožňuje vyvíjanie VR aplikácie pre viac druhov značiek VR setov.

Prispôbenie VR pre aplikáciu a používateľa

Pri navrhovaní VR aplikácie je teda nutné zohľadniť všetky situácie, ktoré už nie sú automaticky implementované týmito frameworkami (či už pre 3D alebo VR). Hlavný príklad je potreba vytvoriť pre každý funkčný 3D objekt (užívateľ má mať možnosť s ním interagovať) obsiahnutý v aplikácii implementáciu, ktorá okrem hlavného prípadu použitia aj eliminuje nežiaduce vedľajšie efekty. Pre príklad je v nasledujúcom zozname zopár potenciálnych nežiadúcich stavov:

- zabránenie užívateľovi prechodu cez 3D objekt (fyzická hranica užívateľ–objekt)
- podmienené povolenie/zabránenie prekryvu dvoch 3D objektov

- aktivácia/deaktivácia fyzikálneho správania objektu (fungovanie gravitácie a pod.)
- rozličné fyzikálne nastavenia častí kompozitného objektu (objekt zložený z viacerých častí)

Ďalší prípad, pri ktorom je potrebné zvážiť spôsob implementácie je dovolený spôsob pohybu vo VR aplikácii. Všeobecne máme možnosť využiť tri varianty pohybu:

- prirodzený pohyb užívateľa v realite
- teleportácia užívateľa a otáčanie užívateľa pomocou tlačidiel a joysticku na kontroléri
- kontinuálny pohyb a otáčanie užívateľa pomocou joystickov na kontroléri

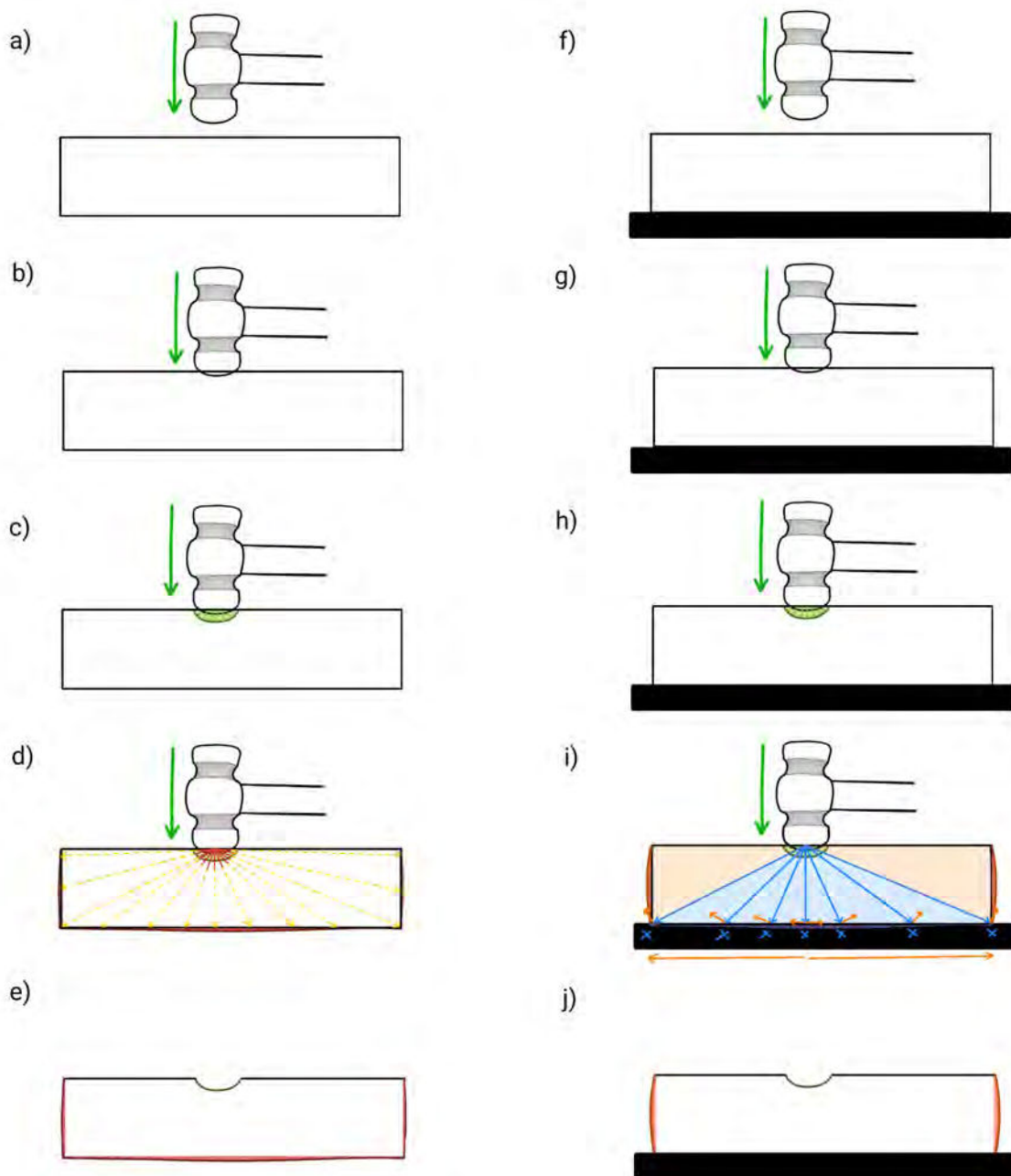
Každá varianta poskytuje iné výhody a nevýhody a sú vhodné pri rozličných situáciach. Teleportácia sa najviac hodí do veľkých otvorených virtuálnych priestorov, ktoré sú mnohonásobne väčšie ako dostupný priestor v miestnosti, v ktorej sa nachádza VR set (bez ohľadu na spôsob snímania polohy). V takomto otvorenom virtuálnom priestore by užívateľovi trvalo pridlho sa niekam dostať za pomoci kontinuálneho pohybu, a bolo by nemožné použiť prirodzený pohyb, keďže by si užívateľ mohol ublížiť. Taktiež je potrebné vziať do úvahy nárazový presun a instantnú zmenu zorného poľa pri teleportácii, čo pri dlhodobom používaní VR aplikácie môže niektorým ľuďom spôsobovať nevoľnosti. Pre potreby navrhovanej aplikácie je teda zohľadnená veľkosť scény, aby užívateľ nemusel vykonávať veľa umelého pohybu v scéne aplikácie. Vďaka tomu môže využívať prirodzený pohyb, keďže všetky potrebné predmety a zariadenia by mal užívateľ priamo po ruke a bude stačiť, aby sa iba otočil.

V aplikácii je možné chytiť a manipulovať predmetmi pomocou ovládačov. Tie slúžia ako interaktory, a ostatným predmetom v scéne je potrebné nastaviť príznaky, ktoré by umožnili ovládačom predmet uchopiť do ruky. Väčšina objektov je navrhnutých ako statická, pretože sa jedná o veľké objekty ako nábytok a nehybné kováčske nástroje. Hlavné predmety určené pre tzv. *grabbing* sú kovové deformovateľné objekty a kladivá. Taktiež je potrebné zohľadniť predvolené chovanie uchopenia VR ovládačmi, pretože predmet po uchopení sa premiestni vycentrovaný do miesta, kde má užívateľ ruku. Takáto funkcionálna pôsobí neprirodzene a riešením je upraviť toto správanie vlastnou implementáciou. Ďalšou prenastaviteľnou konfiguráciou je pohyb uchopeného objektu v priestore. Predvolené nastavenie spôsobuje oneskorené posunutie predmetu, čo pri VR zariadeniach so slabším hardwarom môže pôsobiť vizuálne nepríjemne. Preto je vhodné takéto nastavenie pozmeniť.

Ďalšie nastavenie, ktoré je potrebné prispôbiť pre účely aplikácie je umožnenie predmetu, aby malo fyzikálne vlastnosti v pokojovom stave a aj počas uchopenia. Príkladom je kladivo – ako nestatický predmet by mal byť ovplyvňovaný fyzikou, napríklad gravitačnou silou. Uchopenie kontrolérom toto správanie dočasne vypne, avšak vedľajší efekt spôsobí to, že sa úder kladiva do kovového objektu nezaznamená. Ak by však kontrolér ponechal fyzikálne nastavenia, nebolo by ho možné zdvihnúť. Riešenie predstavuje špeciálne rozdelenie 3D objektu na menšie podčasti, na ktoré sa selektívne aplikujú zmeny nastavené uchopením predmetu do ruky. Podobná situácia nastáva pri interakcií s kovovým objektom. Pokiaľ je kov v pokojovom stave na odkladových plochách, má fyzikálne vlastnosti, okrem situácie kedy je uchopený, alebo položený na nákovu. Kovový objekt s fyzikálnymi vlastnosťami nemôže byť deformovaný, preto je toto správanie v okolí nákovy upravené.

6.5 Deformácia materiálu

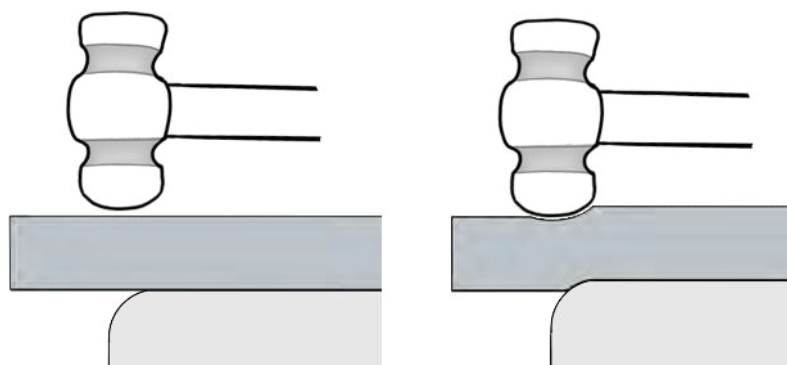
Deformácia kovu patrí medzi deformácie plastické, nie elastické. To znamená, že deformované časti sa po uvoľnení nevracajú do pôvodného tvaru ale ostávajú deformované. Kov si pri kutí z väčšej časti udržiava svoj objem a iba jemne z neho ubúda, tak by to malo byť aj po deformácii v aplikácii.



Obr. 6.18: Vľavo postup riešenia deformácie od kladiva pri zanedbaní nákovy. c) primárna deformácie, d) sekundárna deformácia. Podobne napravo deformácia na nákovy.

Základná deformácia je v oblasti, kde kladivo udrie kov. Kov by sa mal deformovať podľa tvaru kladiva (ako kladivo berieme povrch hlavy kladiva, konkrétne časť určenú na deformovanie kovu). Čím väčší má kladivo povrch, tým plytšie deformácie vytvára. Naopak, zúžená hlava kladiva vytvára hlbšie deformácie. Je teda potrebné, aby sa po kolízii kladiva s kovom kov deformoval na základe tvaru kladiva (obr. 6.18 b, c). Tento typ deformácie je možné dosiahnuť tým, že sa zistia všetky body pri kolízii a transformuje sa ich pozícia na základe smeru úderu a smerov podľa rozpínavosti kovu. Je však nutné určiť akým spôsobom by sa mali transformovať. Okrem typu kladiva závisí taktiež na váhe kladiva a sile úderu. Podľa týchto parametrov je možné zistiť nárazovú silu v mieste úderu (kolízie) a na základe toho deformovať povrch kovu. Keby bola použitá iba táto deformácia, zmenšil by sa obsah kovu. Kov, ktorý bol stlačený kladivom, sa musí niekam presmerovať, a teda do bokov a do dola v smere úderu kladiva. Keďže celá hmota sa posúva od stredu kolízie do priestoru, navrhnutá bola možnosť posúvania jednotlivých vrcholov od stredu kolízie – kolízneho bodu. Tento bod, umiestnený na povrchu kovu by ovplyvňoval body okolo neho iba v tej istej rovine, preto sa bod odsadí od povrchu v smere opačnom ku úderu kladiva. Vďaka tomu budú všetky vrcholy posúvané od tohto bodu (v okolí kolízie do vnútra kovu, a ostatné vrcholy od kolízneho bodu do strán) (obr. 6.18 b, d).

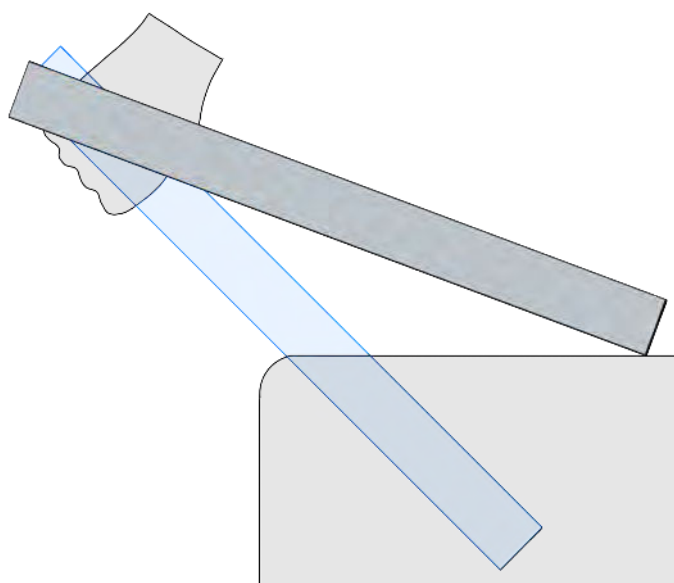
Kov sa nedeformuje vo vzduchu, ale je položený na nákovu, ktorá ho taktiež ovplyvňuje. Ak je kov položený na celej ploche nákovy, nemôže sa rozpínať nadol, môže sa rozpínať iba do bokov (obr. 6.18 g, i). Ak ale je položený čiastočne na nákovu a čiastočne mimo nej, po údere z vrchu môže časť kovu ostať zastavená o nákovu a časť kovu pokračovať v smere úderu do voľného priestoru popri hrane nákovy (obr. 6.19). Podobne to platí aj pre ohýbanie kovu na rohu nákovy.



Obr. 6.19: Deformácia kovu na hrane nákovy pomocou kladiva (vľavo pred deformáciou, vopavo po deformácii)

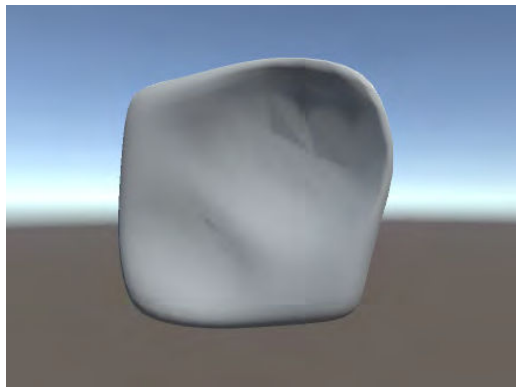
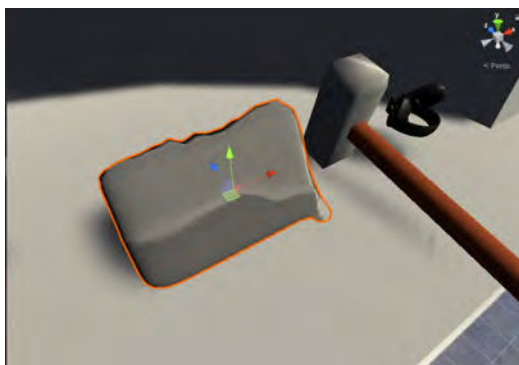
Problémy vznikajúce pri deformácii môžu byť spôsobené pri detekcii kolízie. Kolízia nie je detekovaná pre dve kinematické telesá (telesá, ktoré majú vypnutú fyziku v rigid-body). Objekty sú nastavené ako kinematické, ak sú chytené rukami aby sa pohybovali s rukami a neboli ovplyvnené fyzikou (teda nepadali a nesnažili sa pohybovať). Riešením by mohlo byť nastaviť hlavu kladiva ako nekinematickú a zároveň nastaviť aby nebola vtedy ovplyvňovaná gravitáciou, iba rukoväťou kladiva (tá je ovládaná rukou, respektíve ovládačom). Ďalšou možnosťou by mohlo byť vytvorenie neviditeľných duplicitných kolíznych telies, ktoré by neriešili fyziku objektu, ale iba kolízie. Navyše stále je problematické riešiť naraz s tým zamedzenie kolízie s nákovou v momente úderu kladiva. To môže spôsobiť problémy, kedy by sa malo deformovať kov, ale polovica objemu by kolidovala a musela byť

presunutá mimo nákovu. Ošetrovanie tohto problému je možné napríklad pomocou inšancovania kovového objektu pri kolízii s nákovou. Pôvodný kovový objekt by sa choval stále rovnako, pohyboval by sa s rukou, v ktorej je uchopený a bol by priesvitný (obr. 6.20). Nový inšancovaný kov s fyzikálnym nastavením je kovadlinou vytlačený von mimo kolízie. Problém nastane pri udretí kladivom do tohto inšancovaného kovového objektu. Keďže má fyzikálne vlastnosti, a nie je nijak zastavený alebo držaný, tak je ovplyvnený silou kladiva a odletí od nákovy. To taktiež nie je žiaduce. Ak by žiadna z týchto možností nebola vhodná, je možné vytvoriť skript s iným algoritmom pre detekciu kolízií pre jedno až dve pohyblivé telesá a jedno nepohyblivé. Navyše, aj keď je použitý mesh collider, aby presne definoval povrch kovového objektu tak ak je nastavený ako konvexný, môže sa správať fyzikálne ale bol by problém s deformáciou. Kov má však konvexný tvar iba na začiatku, už po prvom údere sa vytvárajú konkávne časti. Pri konkávnom meshi sa nedá využívať fyzika na ovplyvňovanie, ani zisťovanie kolízií.



Obr. 6.20: Pôvodný kovový objekt je znázornený priesvitnou modrou držaný v ruke a inšancovaný šedý kov môže byť ovplyvnený fyzikálnymi silami.

Otázkou je taktiež spôsob deformácie meshu. Kov sa deformuje na niektorých častiach podobe ako pri soft body, keďže je deformovaný iným telesom a rozpína sa do voľných smerov (obr. 6.22, vpravo). Na rozdiel od soft body ale nie je elastický, takže sa nevráti do pôvodného tvaru. Takže vnútorné pružiny podporujúce tento efekt by museli byť buď zrušené alebo po deformácii upravené aby základné nastavenie bolo to po deformácii. Ďalšia možnosť by bola využiť časticový systém, poprípade využiť krivky pre popis povrchu kovového objektu.



Obr. 6.21: Ukážka subtraktívnej deformácie objektu po interakcii s kladivom vľavo. Deformácia objektu na základe použitia sily od miesta kolízie, podobná ako pri soft-body.

Alternatívne spôsoby deformácie so zachovaním objemu telesa

Pre zlepšenie alebo zrýchlenie počítania deformácii boli študované algoritmy alebo systémy, pripomínajúce čiastočné problémy aplikácie. Deformáciu nahriateho kovu, ktorý by mal zachovávať objem podobne ako pri *soft-body* telesách. ďalším príkladom je riešenie fyzikálnej vlastnosti deformácie pomocou častíc. Popríklad adaptácia systému VIPER na popis meshu a možnosti predlžovať alebo skracovať objekt, zatiaľ čo objem zostane rovnaký.

Všetky tieto algoritmy sú prispôbené pre konkrétny problém alebo funkcionality, ktorú majú riešiť. Preto pre problém kováčskej deformácie tuhého telesa by bolo potrebné ich výraznejšie upraviť a zrušiť tak ich pôvodnú funkcionality. Každý z nich má špecifické problémy pre túto prácu.

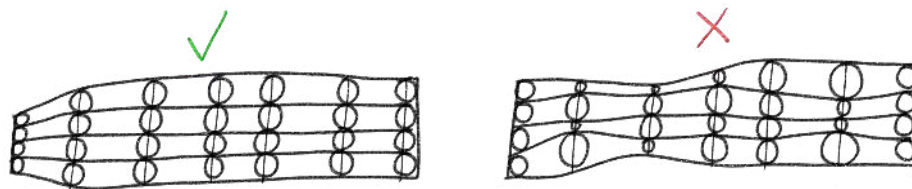
Soft-Body – Môže zachovávať objem ale muselo by sa upraviť aby sa po deformácii objekt nemal tendenciu vrátiť do pôvodného stavu. K tomu by bolo potrebné určiť, kedy sa ešte majú vykonávať kroky deformácie a kedy ich už zastaviť. Podobný prístup je možné použiť na tenké kovové objekty akými sú plechovky. Nie je úplne vhodný pre deformácie pevných telies.

Particles – Pre vyriešenie fyzikálneho roztahovania kovu pri deformácii by bolo možné použiť častice uzatvorené v objekte². Pri zatlačení na tento objekt sa častice posúvajú od tohto miesta preč, lebo nemôžu kolidovať s týmto objektom (podobne ako voda obteká prekážky alebo po zatlačení steny prispôsobí svoj tvar zatiaľ čo si zachováva objem. Problém by mohol nastať znova pri definícii kedy sa má objekt stlačiť a zostať v tej pozícii aj po uvoľnení tlaku/úderu iného objektu. Chovanie by bolo podobné ako balónik naplnený vodou poprípade tuhšou kvapalinou.

VIPER (angl. *Volume Invariant Position-based Elastic Rods*, voľne prevzaté z [4]) je model, ktorej štruktúra predstavuje prút zložený z uzlov, ktoré majú svoj objem a sú navzájom prepojené vektormi. Takýto prút má svoj objem, ktorý si dokáže zachovať pri aplikovaní deformácií, a zároveň vektory umožňujú štruktúre nadobudnúť rozličný tvar. Pomocou VIPER štruktúr by bolo možné vytvoriť model kovového objektu, pomocou rozdelenia meshu na rovnomerné časti. Tie rozdelí podľa hrúbky a nakoniec vzniknú prúty s objemom podľa tohto rozdelenia. Predstavuje to jednoduchší popis meshu objektu ktorý vďaka štruktúre VIPER je možné roztahovať a stláčať, zatiaľ čo si zachováva svoj objem. Keďže ale pri kutí meču je objem v priereze často veľmi odlišný (ostrá hrana prechádza do hrubšej časti meča,

²http://mmacklin.com/pbf_sig_preprint.pdf

následuje ryha ako zúžená časť a znova hrubšia časť a nakoniec znovu ostrá hrana), tak ho nie je možné popísať takýmto systémom. Objekt by musel byť rozdelený nerovnomerne, čo by znemožnilo použitie štruktúr VIPER.



Obr. 6.22: Vľavo normálne použitie systému VIPER pre deformáciu časti meshu, vpravo ukážka meshu ktorý by bolo potrebné popísať ale nie je vhodné pre VIPER. Krúžky predstavujú guľové telesá zobrazujúce objem meshu a vodorovné pásy sú krivky s objemom určeným pomocou guľových telies.

6.6 Doplnkové efekty

Pre dotvorenie užívateľského zážitku z aplikácie sa dá pridať rôznorodé množstvo detailov. Jedným z nich je tvorba realisticky vyzerajúceho ohňa v peci. Keďže oheň ako taký sa veľmi rýchlo náhodne vizuálne mení, nedá sa vytvoriť ako objekt s materiálom. Na takýto druh efektu sa dajú použiť častice (angl. *particles*), ktoré sú schopné napodobniť práve vizuálne správanie ohňa.

Možnosť, ako dodať užívateľovi spätnú väzbu pri virtuálnom kovaní je viac. S ohľadom na spôsob realizácie prichádzajú do úvahy dve možnosti – zvuk a haptická odozva. Zvukový vnem predstavuje doplnkovú informáciu pre užívateľa, nie je zásadne vyžadovaný pre správne pracovanie s aplikáciou, skôr na spríjemnenie a dotvorenie realistického zážitku. Je vhodný na potvrdenie odozvy pri kovaní, čím si užívateľ môže potvrdiť že jeho úder je dostatočne silný a kov skutočne deformuje. Okrem toho sa dá použiť zvuk napríklad pre pec a ozvučenie ohňa rôznymi praskavými zvukmi. Takýto zvuk sa potom nastaví ako priestorový a môže zlepšiť orientáciu užívateľa v priestore virtuálnej kovárne.

Spätná väzba pomocou haptických signálov do ovládačov sa tiež dá využiť pre dodanie realistického detailu. Zaslaním signálu na rozvibrovanie kontroléra pri konkrétnej interakcii sa môže dať užívateľovi najavo nejaká konkrétna informácia. Priame využitie haptickej odozvy je použité napríklad počas udierania kladiva do kovového materiálu poprípade na upozornenie nesprávneho umiestnenia kovového objektu na nákovu.

Pre vytvorenie realistickjšieho dojmu z aplikácie nie je pridané grafické užívateľské rozhranie ani tlačítka napríklad na ovládanie zapnutia pece, poprípade vygenerovanie nového kovového objektu. Všetka vizualizácia, ovládanie a zvláštne efekty sa snažia priblížiť aplikáciu realite.

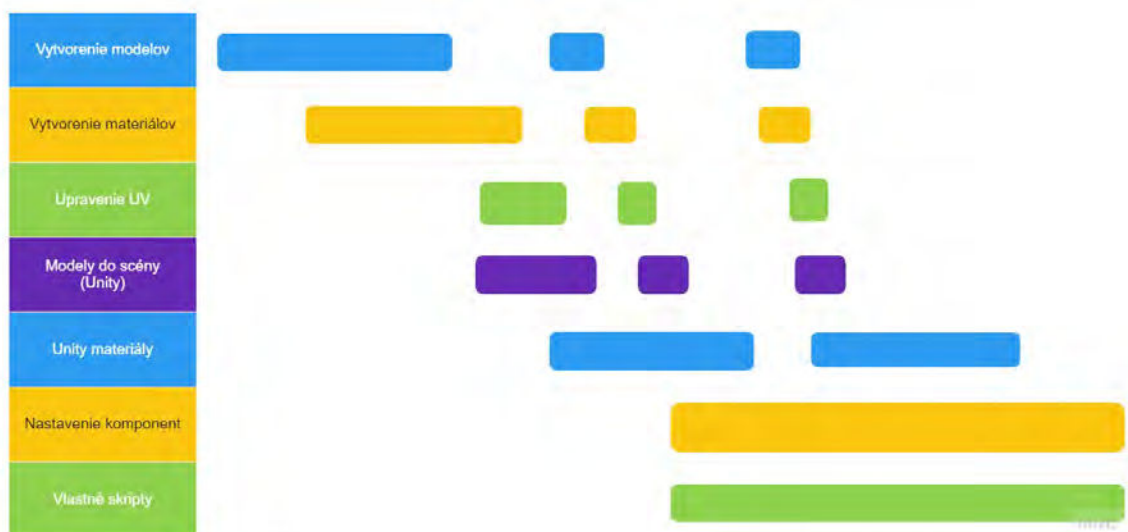
Kapitola 7

Implementácia

V rámci tvorby tejto aplikácie sú použité najmä programy Blender a Unity. Vo frameworku Unity bol využitý jazyk C# pre užívateľské skripty a nástroj Shadergraph pre tvorbu procedurálnych materiálov. V tejto kapitole je popísaná tvorba jednotlivých častí a ich následné zostavenie do funkčného celku. V nasledujúcich sekciách je popísaný postup tvorby modelov, vytvorenie základného 3D VR prostredia, vloženie modelov do aplikácie, ich konfigurácia a naimplementovanie ich správania. Pri širokej dostupnosti rozmanitých VR setov boli použité technológie umožňujúce naprogramovanie aplikácie tak, aby fungovali na rôzne sety. Z dôvodu dostupnosti však bolo možné použiť HTC Vive alebo Oculus Rift (obr. 5.8 prevzaté z [2]). Oba tieto headsety využívajú senzory mimo headset, nazývané outside-in. HTC Vive používa 2 kocky ako outside-in senzory. Tieto senzory musia byť umiestnené v rohoch v miestnosti – hracej plochy – na diagonále tak, aby boli presne nasmerované na seba aj na hernú plochu a zároveň boli dostatočne vysoko (vhodné je aspoň 2m nad zemou), aby dokázali snímať celú plochu. Oculus Rift používa tiež dva snímače, ale tie majú byť na jednej strane (väčšinou na strane, kde je položený počítač), najlepšie okolo dvoch metrov vzdialené od seba. Tieto snímače nepotrebujú vidieť na seba navzájom a hraciu plochu vymedzujú podľa spojeného priestora, na ktoré snímače vidia, navyše nie je treba ich montovať vysoko ako pri HTC Vive, je možné ich položiť aj do výšky priemerných stolov. Je teda omnoho jednoduchšie vytvoriť podmienky pre použitie Oculus Rift (v menších priestoroch a bez stojanov či vysoko umiestnených poličiek). Z hľadiska samotného použitia sa ale tieto headsety veľmi nelíšia. Z hľadiska vytvorenia VR aplikácie v Unity, oba headsety majú podporu, aj keď oficiálne je podporovaný práve Oculus.

Pri implementácii takéhoto softwaru je potrebné použiť rozličné postupy pre vytvorenie dát, ktoré sú vyžadované pre finálne naprogramovanie aplikácie. Jej tvorba teda zahŕňa výrobu modelov, ich metadát a konvertovanie medzi prostrediami, ich programovanie a ladenie. Z toho dôvodu pri opakovaných úpravách je potrebné vykonávať znovu tieto činnosti a vzniká pomerne komplikovaná réžia. Približný postup vývoja takéhoto druhu aplikácie znázorňuje diagram 7.1 (pre vizualizáciu bol tento postup zjednodušene znázornený, no celkový postup bol viac komplexný). Vytváranie modelov – 3D objektov v programe Blender predstavuje počiatočný krok, keďže bez nich by nebolo možné zostaviť scénu v Unity. Tvorba materiálov je separátna časť, z dôvodu rôznych potrieb na jej vytvorenie. Upravovanie UV je závislé na predošlých dvoch krokoch tak, že je nutné spracovať túto časť implementácie vždy až keď daný objekt v Blendery má model a materiál. Vkládanie modelu do scény môže prebehnúť aj pred úplným doladením materiálu a UV. Pre importovaný model sa musí materiál ručne prekonvertovať, čo spôsobuje rozličná reprezentácia dát modelov v programe Blender a v systéme Unity. Počas, a ideálne po konečnom pridaní modelu, materiálu a všet-

kých vizuálnych metadát do Unity zostáva nadefinovať komponenty novopridaných herných objektov, tvorba a ladenie užívateľských skriptov. Tento proces zobrazený na diagrame sa opakuje ľubovoľný počet krát a závisí na počte 3D objektov, ktoré majú v aplikácii figurovať. Pre statické modely, ktoré v aplikácii nevykonávajú žiadnu konkrétnu rolu, je potrebné vykonať kroky 1–4.



Obr. 7.1: Diagram znázorňujúci postup realizácie implementačných častí. Farebné obdĺžniky znázorňujú časové úseky postupu, ako dlho jednotlivé časti približne trvali, ako sa navzájom prelínali a ako na sebe jednotlivé časti záviseli.

7.1 Tvorba 3D objektov

Do scény je potrebné vložiť objekty pre vytvorenie prostredia kovárskej dielne a samotné nástroje použiteľné v aplikácii. Komplexné objekty je náročnejšie vytvárať rovno v Unity, a preto je lepšie použiť externý program Blender na túto úlohu. Táto aplikácia je open-source a vhodná na modelovanie a sculpting rôznych typov objektov, animácie, video, vytváranie textúr ale aj procedurálneho materiálu. Objekty, ktoré je vhodné vytvoriť pre scénu sú pec, nákova, kladivo a samotný kov, ktorý bude deformovaný.

Objekty sú z väčšej časti neorganické a preto je použité modelovanie meshu zo základných prvkov vrcholov, hrán a strán (*point*, *line*, *plane*). Tie sú upravované do výsledného tvaru použitím základných operácií pre úpravu, akými sú úprava mierky (*scale*), posun (*move*) objektu alebo jeho častí (vrcholy, hrany, strany), vysunutie (*extrude*), rozdelenie pomocou slučky (*loop cut*), rozdelenia meshu (*subdivision surface*) a ďalšími.

Hotovému modelu sa vytvorí materiál v záložke tieňovania (*Shading*) pomocou nastavenia typu materiálu a úprave jeho hodnôt akými sú farba, metalickosť, hladkosť povrchu, odlesky, normály a ďalšie. Do týchto parametrov je možné vložiť textúry, ktoré sú komplexnejšie než jednoduchá farba a dokážu sa viac priblížiť k hodnovernejšiemu výzoru. Materiály je tiež možné vytvoriť procedurálne zložením logických operácií, parametrov a rôznych funkcií. Pre vytvorenie realistickejších materiálov procedurálnym spôsobom sa využívajú napríklad šum (*noise*), vlnenie (*wave*), prípadne *voronoi*. Na rozdiely blízko hrán a prechody medzi textúrami a vzormi sa Využíva tiež zmiešanie farieb (*color ramp*) a pre vytvorenie 3D efektu sa používa *bump*.

Všetky tieto postupy je možné skombinovať a prispôbiť si materiál podľa predstáv. Pri použití 2D textúr pre vytvorenie materiálu je väčšinou potrebné upraviť namapovanie textúr pomocou UV. Vtedy sa 3D mesh objektu rozprestrie do 2D priestoru a na základe toho sa dá namapovať časť textúry na danú časť modelu.

Všetky objekty okrem kovu sú statické z hľadiska meshu. Keďže sa netransformujú, je možné vytvoriť k nim textúry klasickým spôsobom, čiže nie je nutné ich tvoriť procedurálne. Na vytvorenie textúr je možné vyžiť textúru kovu, poprípade aj normálové mapy, výškové mapy alebo iné. Tie je buď nutné stiahnuť alebo vytvoriť vlastné.

Doladené modely sú exportované vo formáte *fbx*, ktorý obsahuje mesh objektu a základný materiál. Z dôvodu nekompatibility systémov Unity a Blender nie je možné využiť väčšinu materiálov z aplikácie Blender. Možnosťami sú teda upraviť jednotlivé hodnoty ktoré nie sú rovnako ukladané v Unity a v Blenderi (napríklad *Smoothness* z Blenderu je v Unity významovo podobné s *Roughness*, a dá sa upraviť ako $1 - \text{Smoothness}$). Niektoré nastavenia a parametre však nie je možné tak jednoducho upraviť. Textúry musia byť upravené a pregenerované, aby ich bolo možné použiť v Unity. Procedurálne vytvárané textúry pomocou Blender blokov (*node*) nie je možné preniesť do Unity. Preto je nutné buď procedurálne materiály uložiť ako 2D textúry pomocou funkcie *bake*, čím stratia svoju procedurálnu funkciu, napríklad pohyb v čase alebo pri úprave meshu objektu. Ďalšou možnosťou je vytvoriť podobný procedurálny materiál v aplikácii Unity pomocou funkcie *ShaderGraph*. Tá je však dostupná iba pri použití *URP* – renderovacieho systému v Unity.

Procedurálne vytvorený materiál je potrebný pre vizualizáciu nahrievania kovového objektu a jeho deformácie. Ďalej je použitý pre vizualizáciu vody, ktorá sa pohybuje v čase. Môže taktiež zlepšiť výzor jednoduchých materiálov poprípade materiálov z 2D textúr.

7.2 Unity Setup

Dôležitým krokom pri tvorbe aplikácie virtuálny kováč je správne zostaviť vývojové prostredie. Pre implementáciu a preloženie zdrojových kódov, modelov a konfigurácii je využité prostredie Unity. Základom tohto prostredia je hierarchia objektov, z ktorých potom bude tvorený 3D priestor viditeľný cez VR set. Koreňový objekt predstavuje scénu, do ktorej sú vkladané ďalšie objekty. Najdôležitejší objekt v tejto scéne predstavuje VR set umožňujúci interagovať s týmto 3D prostredím.

Package Manager

Súčasťou Unity je tzv. **Package Manager**, ktorý figuruje ako verejný repozitár s rôznymi rozšíreniami. Tieto rozšírenia môžu predstavovať rôzne pomôcky a nástroje pre Unity editor ako taký, alebo poskytujú knižnice s danou špeciálnou funkcionalitou zjednodušujúcu programovanie aplikácie. Pre potreby vytváranej aplikácie sú využité rôzne vstavané balíky, najdôležitejšie z nich sú **Universal Rendering Pipeline (URP)** a **XR Interaction Toolkit (XR)**. URP je základný Unity balík upravujúci renderovanie preloženej aplikácie. Táto verzia rendering pipeline umožňuje využívanie nástroja Shadergraph, vďaka ktorému možno tvoriť procedurálne materiály. XR je pre aplikáciu kľúčová knižnica, ktorá zapája do Unity funkcionalitu virtuálnej reality.

XR Interaction Toolkit

Na vyvíjanie na VR v Unity je potrebný vstavaný balík umožňujúci zobrazenie a interakciu vo VR zariadení. Pre tieto účely sa využíva multiplatformný balík **XR Interaction Toolkit**. Jedná sa o komponentovo založený systém, ktorý umožňuje prepojenie vstupov v Unity frameworku s virtuálnou realitou. Funguje pre rôzne VR platformy, ako napríklad **Meta Quest**, **OpenXR**, **Windows Mixed Reality**, atď. Jeho hlavnou základnou funkcionalitou je začleniť VR set ako vstupno-výstupný prvok v Unity aplikácii, teda interakcia s predmetmi v aplikácii pomocou kontrolérov, spätné reakcie pomocou haptických impulzov (vibrácie) a vizuálny výstup do headsetu. Ten je realizovaný pomocou **XR Origin**.

Jadrom tohto systému je dvojica komponentov **Interactor** a **Interactable**. Tie sú prepojené pomocou **Interaction Manager**, ktorý spravuje udalosti vytvorené medzi týmito dvoma komponentami. Okrem toho obsahuje ďalšie komponenty, ktoré sú použiteľné pre pohyb a vykresľovanie vizuálov v aplikácii.

Na využitie alebo rozšírenie funkcionality sa používajú skripty a predpripravené prefaby umožňujúce využitie kamier v headsete alebo celej štruktúry **XR Origin**, umožňujúcej pohyb a použitie ovládačov. **XR Origin** je vhodnejší pre použitie, pretože obsahuje okrem kamery separátne pre každé oko aj nastavenie častí v štruktúre a ovládače určené na interakciu s prostredím. Pre možnosť pohybu sa využíva časť **Locomotion**, v rámci ktorého je možné nastaviť rôzne typy pohybu a otáčania. Ďalšia veľmi dôležitá súčasť **XR Toolkitu** je **XR Grab Interactable**, ktorá je určená na uchopenie predmetov v scéne pomocou ovládačov. Táto funkcionalita je dôležitá pre manipuláciu objektov v scéne.

Niektoré predvolené nastavenia bolo následne potrebné trochu upraviť pre zlepšenie užívateľského zážitku. Jednou z nich je úprava locomotion system objektu v scéne tak, aby bolo možné joysticky kontrolérov využívať na plynulé otáčanie a pohyb po prostredí oproti pôvodnému teleportačnému spôsobu, ktorý posúval užívateľa o fixný uhol a vzdialenosť.

Ďalšou dôležitou úpravou je zmena spôsobu úchyty predmetov. Predvolený spôsob naimplementovaný v **XR Grab Interactable** zvolí vždy jeden konkrétny bod úchyty (*attach*

point), v ktorom sa potom zobrazuje vizuálny model ruky (kontroléru) pri manipulácii s držaným predmetom. Takýto spôsob má rôzne limitácie, ako napríklad nemožnosť chytiť kladivo z rôznych smerov, alebo kovový materiál za ľubovoľnú časť, a tým pádom by bolo veľmi ťažké s ním hocijako manipulovať. Riešením je pridanie skriptu `OffsetGrab`, ktorý rozširuje funkcionality `XR Grab Interactable` a umožňuje uchopiť predmet v hociktorej časti objektu, pokiaľ sa nachádza v kolíznej oblasti kontroléru s daným objektom. Vďaka tomu je manipulovanie s objektami oveľa realistickejšie. Okrem rozširovania funkcionality je potrebné pri chytateľných objektoch nakonfigurovať aj spôsob pohybu držaného objektu. Táto knižnica predvolene realizuje pohyb predmetmi kinematickým spôsobom, teda objekt sa hýbe za polohou kontroléra. Pri spustení aplikácie však toto nastavenie vytvára dojem sekania obrazu, čo nevlplyva dobre na oči užívateľa. Preto je pri väčšine chytateľných objektov nastavený okamžitý (angl. *instantaneous*) pohyb, čím vyzerá manipulovanie s objektami prirodzenejšie.

7.3 Herné objekty

Herné objekty (prekl. z triedy `GameObject` v aplikačnom rozhraní Unity frameworku) sú teda základné stavebné kamene umiestnené v scéne aplikácie. Tá obsahuje úplne všetky elementy tvoriace prostredie, ktoré vidí a s ktorým môže interagovať užívateľ – podlaha, osvetlenie, budova, kovácke predmety a vybavenie. Základom pre vytvorenie herného objektu je nastaviť vo vývojovom prostredí príslušný 3D model a materiál s textúrou nalinkovaním príslušného súboru obsahujúceho tieto dáta. V závislosti od želaného chovania alebo funkcionality herného objektu je potom nutné pridať a nastaviť ďalšie komponenty.

Komponenty herných objektov

V rámci tvorby herných objektov je potrebné správne nastaviť ich vlastnosti a chovanie. Týmto objektom je možné priradiť komponenty, ktoré dokážu upravovať ich chovanie v 3D priestore a tiež aj ich vzhľad. Do aplikačného prostredia sú dodané buď ako Unity vstavané komponenty (napríklad `RigidBody`, `Collider`), alebo užívateľsky definovanými triedami. Vďaka týmto komponentom možno nakonfigurovať herný objekt tak, aby sa dokázal stať hodnovernou napodobeninou z reálneho sveta.

Tieto komponenty sú naimplementované ako triedy poskytujúce zmienenú funkcionality alebo ovplyvňujúce správanie objektu istým spôsobom. V rámci Unity frameworku sa taktiež nazývajú skripty, avšak dodržiavajú triednu dedičnosť zo základných implementácií z knižnice `UnityEngine`, konkrétne z triedy `Component`¹. Tieto komponenty poskytujú následne rôzne verejné parametre, ktoré môžu byť povinné na vyplnenie pre zaistenie plnej funkcionality daného komponentu. Vyplnením sa rozumie nastavením referencie, ktorú môže predstavovať iný herný objekt, konštantná hodnota, alebo iný komponent z herného objektu. Vďaka tomu napríklad možno konfigurovať a parametrizovať komponenty herných objektov rôznymi konštantnými hodnotami (číselné hodnoty, prípadne výčet konštánt figurujúci ako nastavenie (angl. *Enumeration*)) alebo ich prepojiť s ďalšími komponentami vložením referencie na ne.

Vlastné triedy/skripty upravujúce herný objekt a jeho komponenty môže programátor tvoriť s pomerne malými obmedzeniami. V zásade sú užívateľské komponenty určené na naprogramovanie zmeny správania iných pridaných komponentov herného objektu na základe

¹<https://docs.unity3d.com/ScriptReference/Component.html>

interakcie zo strany užívateľa v aplikácii, alebo nadefinovanie interakcie daného objektu s inými konkrétnymi objektami nejakým konkrétnym spôsobom.

Vstavané Unity komponenty

Tieto komponenty sú súčasťou Unity Enginu. Pre vývojára sú dostupné vo forme aplikačného programovacieho rozhrania (API) a je teoreticky možné ich rozšíriť (napr **OffsetGrab**). Najbežnejšie využívané komponenty pre vytváranie aplikácií boli tieto komponenty:

- Transform
- Mesh Filter
- Mesh Renderer
- Rigidbody
- Collider (Mesh, Sphere, Box, Capsule)

Komponent **Transform** je povinne obsiahnutý v každom hernom objekte a využíva sa na uloženie pozície, rotácie, škály. Bez tohto komponentu by nebolo možné určiť polohu objektu. **Mesh Filter** slúži na priradenie exportovaného 3D modelu objektu a **Mesh Renderer** pre jeho materiál.

Kľúčovým komponentom v rámci tejto aplikácie je využitie **Rigidbody**. Tento komponent umožňuje objektu chovanie na základe fyzikálneho systému Unity Enginu. Vďaka tomu môže byť objekt ovplyvňovaný gravitáciou, alebo silami vyvolanými interakciou s inými objektami. Nieje však vhodné aplikovať tento komponent na objekty ako sú napríklad budovy, pretože to môže vytvoriť nežiadúci efekt, ak by na objekt zapôsobila nečakane veľká sila (v kontexte tejto aplikácie, ak by kladivom bolo udierané do pece, fyzikálny systém by ju posunul, aj keď by sa jednalo o zlomkové hodnoty ktoré nie sú postrehnuteľné voľným okom).

Kolízne komponenty sú neviditeľné objekty priradené k hernému objektu, pričom ich tvar a objem nemusí byť zhodný s jeho skutočným 3D modelom (meshom). Ak sa počas behu aplikácie prekryjú dva herné objekty s Colliderom, Unity systém vyšle do obidvoch objektov správu o kolízii, vďaka čomu možno naimplementovať, ako by mali tieto objekty reagovať na vzniknutú udalosť. Kolízne komponenty sú rozdelené do menších podkategórií podľa ich základného tvaru – **BoxCollider** pre štvorec, **SphereCollider** pre guľu, **CapsuleCollider** pre sploštenú guľu a **MeshCollider** pre nastavenie zložitejšieho tvaru collideru.

Tvorba kováčskych herných objektov

Všetky 3D objekty použité v aplikácii boli ručne vytvorené v programe Blender. Hlavné herné objekty sú vytvorené a nakonfigurované pre účely virtuálneho kovania sú:

- Kovový objekt
- Kladivo
- Nákov
- Pec



(a) Náková



(b) Pec



(c) Kladivá



(d) Stôl



(e) Nádobá s vodou na schladenie kovu

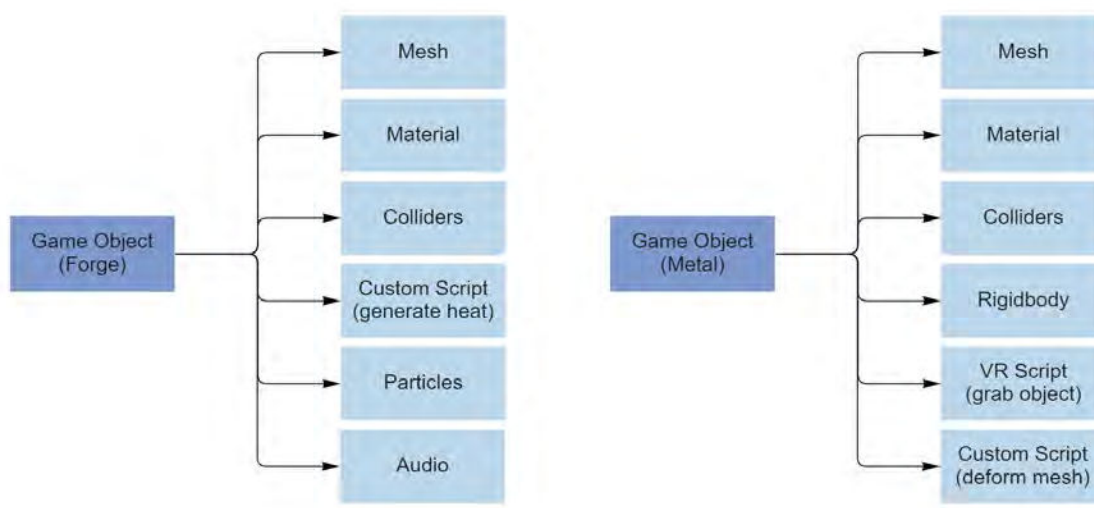


(f) Kovy a kladivá na stole

Obr. 7.2: Ukážka komponent z aplikácie

- Nádobu s vodou

Okrem nastavenia základných komponent sa teda musia pridať komponenty upravujúce interakciu s užívateľom. Niektoré vstavané Unity komponenty riešia požadovanú funkcionality automaticky, niektoré je potrebné dodatočne nastaviť. S pomocou užívateľských skriptov, ktoré môžu byť naviazané na týchto objektoch, je možné nadefinovať ako sa budú v aplikácii správať tieto predmety pri vzájomnej interakcii.



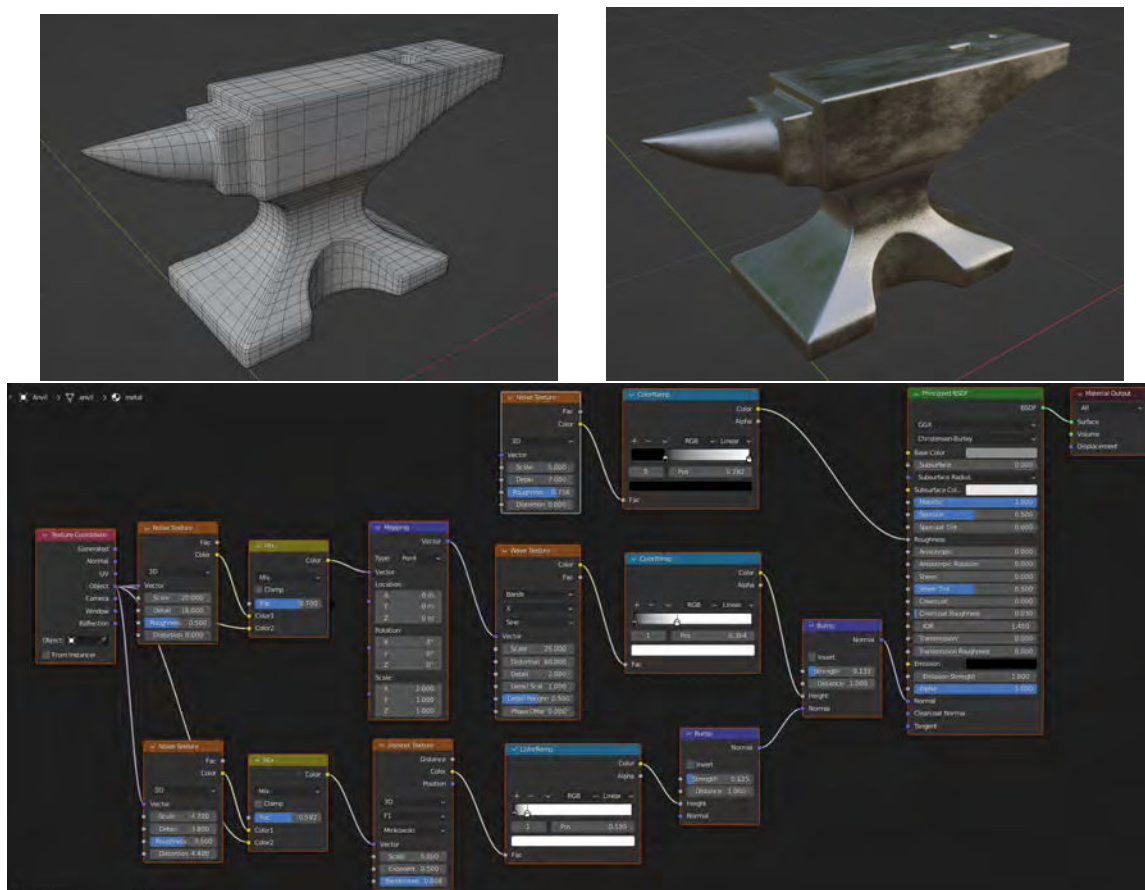
Obr. 7.3: Príklad zoznamu priradených komponentov ku konkrétnym herným objektom – pec a kovový materiál

Obrázok 7.3 zobrazuje zoznamy hlavných komponentov, z ktorých pozostávajú tieto herné objekty. Spoločne aj s ostatnými objektami pozostávajú z 3D objektu (komponenta **Mesh Filter**) a jeho materiálu (**Mesh Renderer**).

Nákova

Pre vymodelovanie objektu nákovy je použitý program Blender (Obr. 7.2a). Modelovaný je pomocou základných modelovacích nástrojov a funkcií ako sú zmena veľkosti (**scale**), posun (**move**) objektu alebo jeho častí (vrcholy, hrany, strany), vysunutie (**extrude**), rozdelenie pomocou slučky (**loop cut**), rozdelenia meshu (**subdivision surface**) s vyhladzovaním meshu a ďalšími (výsledný mesh 7.4, vľavo hore).

Na vizualizáciu je vytvorený procedurálny materiál zobrazujúci kovový povrch nákovy (obr. 7.4, vpravo hore). Používa šum (**Noise Texture**) a vlnovú funkciu (**Wave Texture**) spolu so zmiešavaním, blokom **ColorRamp** a **Bump** pre vytvorenie normál a na definovanie hladšieho alebo drsnejšieho povrchu. Nastavená je aj základná farba, metalickosť a odlesky (ukážka grafu na vytvorenie materiálu pre nákovu 7.4, spodný obrázok).

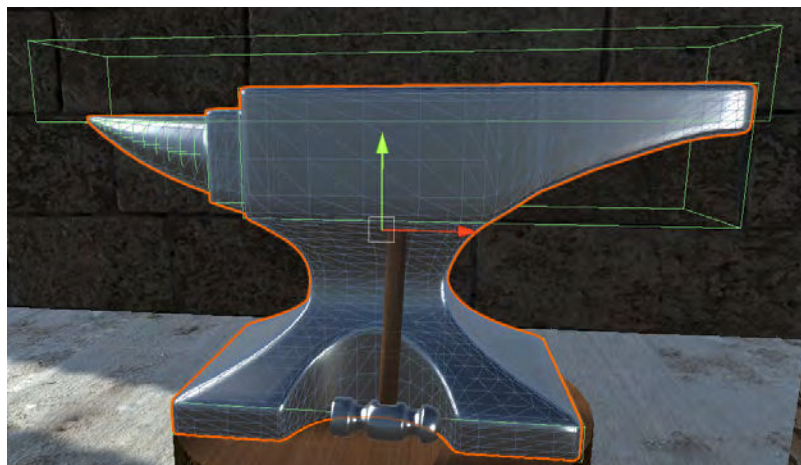


Obr. 7.4: Hore vľavo mesh nákovy, vpravo nákova s materiálom, dole tvorba materiálu pre nákovu

Model je exportovaný vo formáte fbx a nahratý to Unity. Systém ShaderGraph nie je kompatibilný so shading systémom v aplikácii Blender a preto sa nedá použiť procedurálny materiál z Blenderu. Na vytvorenie materiálu v Unity je použitá ocelová textúra². Pre lepšie namapovanie textúr sa upravujú UV v aplikácii Blender.

Nákova je statický (nepohyblivý objekt) požívaný pre deformáciu kovových materiálov. Kolízne telesá teda potrebujú čo najvernejšie popísať tvar nákovy hlavne v miestach kde sa kov deformuje. Na tento účel je najvhodnejšie použiť presný **MeshCollider**. Ten však nie je možné použiť, pretože Unity nedokáže detegovať kolízie medzi dvoma kolíznymi telesami typu **mesh**. Riešením je použitie väčšieho množstva jednoduchých kolíznych telies typu **BoxCollider**, **SphereCollider** a **CapsuleCollider** demonštrovaných na obrázku 7.5.

²<https://www.poliigon.com/texture/metal-stainless-steel-brushed-elongated-005/3189>



Obr. 7.5: Vyobrazenie kolíznych komponent v nákovu

Nákova má taktiež na starosti povolenie deformácii kovového objektu pomocou najvrchnejšieho kolízneho telesa typu **trigger** na povrchu. Pri položení kovu na nákovu je možné kov deformovať. Výnimka je ak je kovový objekt nejakou časťou vložený do samotného tela nákovy, vtedy nie je vhodné aby bol kov deformovaný, lebo by v žiadnom prípade nemal kolidovať s nákovou. Pri pustení kovu v tele nákovy je kov vytlačený z tohto priestoru. Pri vložení kovového objektu do tela nákovy je vytvorená haptická odozva pomocou rozvibrovania ľavého kontroléru. Vďaka tomu užívateľ vie, že kov je držaný alebo umiestnený nesprávne na deformáciu.

Pec

Pec je vymodelovaná v aplikácii Blender (Obr. 7.2b). Použité boli základné úkony **scale**, **move**, **extrude** rozšírené o použitie **loop cut**, úpravu **subdivision surface** bez vyhladzovania a vysunutie pomocou **array**. Pre zjednodušenie vytváraní kolíznych telies v Unity bol model pece rozdelený na jednoduchšie tvary, z ktorých boli neskôr nastavované kolízne telesá.

Na základe jednoduchého materiálu z textúr³ v Blendery je vytvorené UV rozloženie pre správne namapovanie textúr na model. Na exportovanie modelu do Unity je využitý formát **fbx**, ktorý zahŕňa mesh modelu a základný materiál. Pec je v scéne umiestnená rovno pred objekt VR setu aby bola v zornom poli po spustení.

Keďže sa s pecou nepohybuje, je možné použiť parameter **static** kvôli počítaniu osvetlenia a možnosti vytvorenia mapy osvetlenia. Z dôvodu konkávnosti v niektorých častiach, hlavne v otvore na pokladanie nástrojov a nahrievania kovu je zložená z viacerých základných kolíznych telies na základe predpripravených objektov z aplikácie Blender. Tie popisujú tvar pece aby s ňou bolo možné interagovať.

Objekt pece obsahuje taktiež guľové kolízne teleso typu **trigger** v oblasti otvoru pece určené na kontrolu kolízie s kovovými objektami s úmyslom ich nahrievania. Pre vizuálne doplnenie je pridaný časticový efekt znázorňujúci oheň.

Po vložení objektu do pece do ohňa je skontrolovaný a ak je to kovový deformovateľný objekt začne byť nahrievaný. Z dôvodu veľkého meshu ktorý spomaľuje výpočet nie je nahrievaný na každý frame ale iba každý dvadsiaty. Podľa vzdialenosti od centra ohňa sú

³https://polyhaven.com/a/red_bricks_04

časti kovového objektu nahrievané rôznou rýchlosťou. Časť najbližšie ku stredu je nahrievaná najrýchlejšie a časť ktorá je úplne mimo ohňa je ochladzovaná. Ak je kovový objekt úplne mimo pece začne proces chladenia celého objektu (pomalšie než nahrievanie ale rýchlejšie než chladnutie ak je aspoň časť kovového objektu v ohni).

Na dotvorenie zrkovitého a zvukového vnemu je pridané jednoduché bodové svetlo do ohňa a zvuk praskania ohňa. Tento zvuk je nastavený ako 3D, teda čím bližšie je užívateľ ku ohňu, počuje tento zvuk hlasnejšie.⁴

Kladivo

Kladivo je vymodelované z dvoch separátnych častí. Hlava kladiva má dve funkčné strany, ktoré môžu byť rôznych tvarov (Obr. 7.2c). Väčšinou nebývajú obe strany rovnakého tvaru a niekedy nie sú obe priamo používané na deformáciu. V aplikácii je teda vymodelované s jednou oblou stranou a druhou plochou. Plochá strana musí byť stále jemne zaoblená na krajoch aby nevytvárala ryhy ale skôr jemnejšie prechody (príklad neupraveného staršieho kladiva a novšieho zaobleného na obrázku 6.8 vpravo). Rúčka je modelovaná ako sploštené valcovité teleso.

Model má taktiež vyrobený procedurálny materiál v Blendery viditeľný na obrázku 7.6. Pre povrch hlavy je použitý rovnaký materiál ako na nákovce. Pre vizualizáciu rúčky je zložená z podobných blokov ako pre kovový materiál, ale je viac zúžený aby pripomínal drevo. Farby sú vybrané medzi béžovou a hnedou pre zvýraznenie letokruhov.



Obr. 7.6: Kladivo

Po presunutí do Unity, vytvorenie nového materiálu pomocou textúr⁵ a namapovania UV je objekt pripravený na nastavovanie.

Kladivo je dynamický objekt, s ktorým je umožnená interakcia. Užívateľ ho môže chytiť pomocou komponentu `OffsetGrab` (rozšírenie `XR Grab Interactable`) v ľubovoľnom mieste a s ľubovoľným natočením. Ak by bol vymodelovaný ako jeden objekt, po uchopení stratí fyzikálne vlastnosti a nie je možné detegovať kolíziu s fyzikálnymi údajmi potrebnými pre deformáciu. Z tohto dôvodu je rozdelené na dve časti, kedy rúčka zachováva funkciu

⁴Zvuková stopa prevzatá z <https://mixkit.co/free-sound-effects/fire/>

⁵<https://www.poliigon.com/texture/metal-stainless-steel-brushed-elongated-005/3189>, <https://www.poliigon.com/texture/wood-quartered-chiffon-001/3836>

interakcie s užívateľom a hlava kladiva zasa deformačnú funkciu. Kolízne telesá hlavy a rúčky sú k tomu prispôsobené na rôznych vrstvách. Tým sa zaručí, že navzájom nie sú ovplyvňované a môžu správne fungovať. Keďže s objektami je možné manipulovať, alebo samotné manipulujú s inými objektami, oba potrebujú fyzikálny komponent **Rigidbody**. To ale spôsobuje, že sa už nechovajú ako jeden objekt a tak je potrebné použiť **FixedJoint**.

Po náraze kladivom do kovového objektu sa posielajú informácie o sile a smere pohybu kladiva pri kolízii do deformačnej časti pre kov. Ak sú splnené podmienky, je vypočítaná deformácia kovu. Ak je náraz dostatočne silný aby vyvolal deformáciu, prehrá sa zvuk nárazu kladiva⁶ o kovový objekt a odošle sa haptická odozva do pravého ovládača.

Kovový materiál

Kovový materiál je z celej aplikácie najdôležitejší herný objekt (Obr. 7.2f). Vymodelovaný je ako vyhladený kváder rôznych rozmerov. Mesh objektu potrebuje byť dostatočne detailný aby pri deformácii vyzeral prirodzenejšie. Keďže je tento objekt určený na deformáciu dokonca ľubovoľnú, je potreba aby správne detegoval kolízie a na základe nich sa deformoval. Je teda využité kolízne teleso **MeshCollider**.

Deformácia ako taká, a jej výpočet je taktiež ovplyvnený jeho nahriatím. Na to slúži interakcia s objektom pece, kedy vloženie kovového objektu do epicentra ohňa sa začne zahrievať daná časť kovu. Účinok zahrievania je priamo úmerný vzdialenosti k stredobodu objektu ohňa v peci. Pre vizualizáciu urovne nahriatia kovového objektu je vytvorený a použitý procedurálny materiál. Ten je vytvorený pomocou gradientu reprezentujúceho teplotu kovu aplikovaného na vrcholy meshu objektu.

S kovem je možné manipulovať pomocou skriptu **OffsetGrab**. Vďaka tomu je možné ho nastaviť na nákovu do ľubovoľnej pozície pre deformáciu. Ak je však kov vložený do tela nákovy, ľavý ovládač bude vibrovať a tak upozorní užívateľa, že nie je možné kov deformovať. Kov, ktorý je umiestnený správne na nákovu a je dostatočne nahriaty, je možné deformovať pomocou úderov kladivom.

Nádoba s vodou

Nádoba s vodou určená na rýchle ochladzovanie nahriateho kovového materiálu je vymodelovaná ako jednoduchý valec s vysunutou dierou v strede pre vodu (Obr. 7.2e). Nádoba má jednoduché kolízne teleso **BoxCollider**. Mohlo by sa použiť aj iné kolízne teleso z väčšou presnosťou ale to nie je až tak potrebné. Materiál nádoby je vytvorený z textúry drevených dosiek⁷. Hlavnú funkciu zachováva voda. Tá je vymodelovaná ako valec s kolíznym telesom nastaveným na **trigger**. Jej funkciou je zistiť kolíziu s kovovým objektom a ochladiť ho ak táto situácia nastane. Materiál pre tento objekt je vytvorený procedurálne spojením posunu textúry⁸ v čase v smere oproti sebe a podobne aj pre normály. Tento materiál je samozrejme priesvitný.

⁶Stopy prevzaté z <https://www.fesliyanstudios.com/royalty-free-sound-effects-download/hammer-hitting-metal-57>

⁷https://polyhaven.com/a/weathered_brown_planks

⁸https://meocloud.pt/link/8ce20b6d-a1c0-4337-b41a-dea4d89886c4/Water_001_SD/

Podporné objekty

Ďalšie objekty, ktoré boli vymodelované sú napríklad stôl (Obr. 7.2d), peň pod nákovu (Obr. 7.2a), budova s prístreškom, prázdna nádoba (obr. 7.7) a steny. Stoly sú určené na odkladanie nástrojov, kovových objektov a hotových výrobkov, a tak zaručiť jednoduchšiu manipuláciu s nimi. Peň vyvyšuje pozíciu nákovy a netreba sa k nej až moc zohýbať. Budova a steny majú za úlohu dotvoriť scénu a udržať užívateľa na scéne. Ich materiály sú vytvorené pomocou textúr⁹.

Podobne sú vytvorené svetlá napríklad v peci pre oheň alebo pod strechou aby nebolo moc tma, poprípade využitie skyboxu¹⁰.



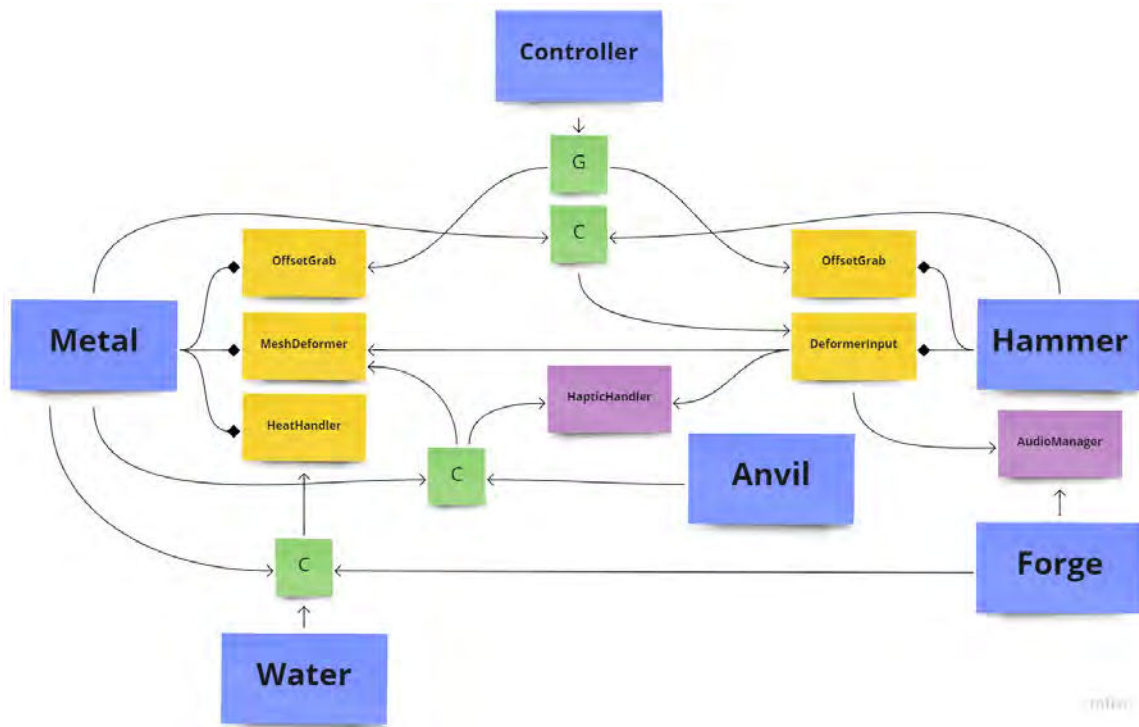
Obr. 7.7: Nádoba na kovové objekty, stôl na ukladanie výrobkov alebo nástrojov a stena za nimi

⁹https://polyhaven.com/a/bark_brown_01, <https://www.poliigon.com/texture/wood-quartered-chiffon-001/3836>, https://polyhaven.com/a/weathered_brown_planks, https://polyhaven.com/a/medieval_blocks_03

¹⁰https://polyhaven.com/a/mossy_forest

7.4 Behaviorálna časť aplikácie

V tejto sekcii sú detailnejšie popísané triedy priradené k herným objektom ako komponenty. Najdôležitejšie komponenty naviazané na objekty sú: `MeshDeformer`, `HeatGenerator`, `DeformerInput`, `HapticHandler`, `AudioManager`, `OffsetGrab`, `MeshGenerator`. Na diagrame 7.8 sú znázornené vzájomné väzby a vzťahy s udalosťami aplikácie a v ďalších podsekciiach sú bližšie popísané najdôležitejšie komponenty.



Obr. 7.8: Herné objekty s užívateľskými skriptami a ich väzbami na ostatné objekty a udalosti. Modré uzly predstavujú konkrétne herné objekty, žlté uzly sú komponenty naviazané na objekty, fialové uzly predstavujú komponenty pripojené na koreňový herný objekt pre účel dostupnosti funkcionality a zelené uzly značia udalosti: **G** = **grab**, **C** = **collision**. Šípky smerujúce do kolízneho uzlu znázorňujú, ktoré dva herné objekty kolidujú a šípka smerujúca z tohto uzlu značí, ktorý komponent spracuje udalosť. V prípade kolízneho uzlu spájajúci **HeatHandler** ide o alternatívu kolízie kovového objektu s pecou alebo vodou. V prípade udalosti uchopenia prebieha spracovanie na respektívnych komponentách **OffsetGrab** na uchopených objektoch.

MeshDeformer

Tento komponent je aplikovaný na všetky objekty deformovateľných kovových materiálov, a implementuje interakciu s hernými objektami – konkrétne s kontrolérmi, kladivami, nákovou, pecou a nádobou na vodu. Tieto objekty invokujú jednotlivé reakcie kovového objektu pri dotyku s triggermi alebo collidermi. Zároveň automaticky spravuje úroveň deformovateľnosti (teplotu) vertexov svojho meshu. Ak sa nachádza v **Collider** komponente pece (s príznakom **trigger**, pričom jeho aktivovanie je spracované v komponente **HeatGenerator**), tak skript priebežne aktualizuje a navyšuje hodnoty po maximálnu hodnotu. Keď kov vložíme do pece, začne sa nahrievať a zároveň meniť farbu. Keď používateľ vloží kov do pece, časť kovu ktorá je vložená v strede ohňa sa nahrieva rýchlejšie než časť ktorá je na okraji. Tieto hodnoty sú využité dvoma spôsobmi – ako výber úrovne gradientu v materiáli spracovanom v **Mesh Renderer**, a tiež ovplyvňuje silu deformácie. Po vybratí objektu z collidera je táto pravidelná aktualizácia invertovaná a zoslabená, čo znamená ochladzovanie kovu (ktoré je oveľa pomalšie ako jeho nahriatie). Ak je však kov iba čiastočne v ohni, časť ktorá je mimo ohňa sa ochladzuje pomalšie než keby bol úplne mimo ohňa. Pre rýchle ochladenie celého objektu je možné použiť nádobu s vodou, pričom pri zaznamenaní kolízie sú hodnoty deformovateľnosti vertexov nastavené na predvolenú minimálnu hodnotu.

Tiež udržiava svoj aktuálny stav deformovateľnosti, keďže to má byť možné len v blízkosti **triggeru** nákovy. Vlastnosti **RigidBody** kovového objektu ovplyvňuje uchopenie (**OffsetGrab**), čím bolo nutné ručne naimplementovať zmenu chovania **RigidBody** aby bolo možné kov deformovať, avšak len v prítomnosti nákovy. Problémom je situácia, keď je kovový objekt zámerne vložený do telesa nákovy. V tom prípade je pri pustení objektu vlastnosť **RigidBody** automaticky prestavená na fyzikálne správanie, a teda je kov vstavaným fyzikálnym systémom vytlačený z nákovy. Nemožno však triviálne zamedziť tomuto prípadu, a ak užívateľ chce správne priložiť kovový materiál k povrchu nákovy pre povolenie deformovania, pri neželanom prekryve s telesom nákovy je vydaný haptický signál do kontroléra, v ktorom užívateľ drží kov. Vďaka tomu dostane spätnú reakciu, aby umiestnil kovový materiál na nákovu správnym spôsobom.

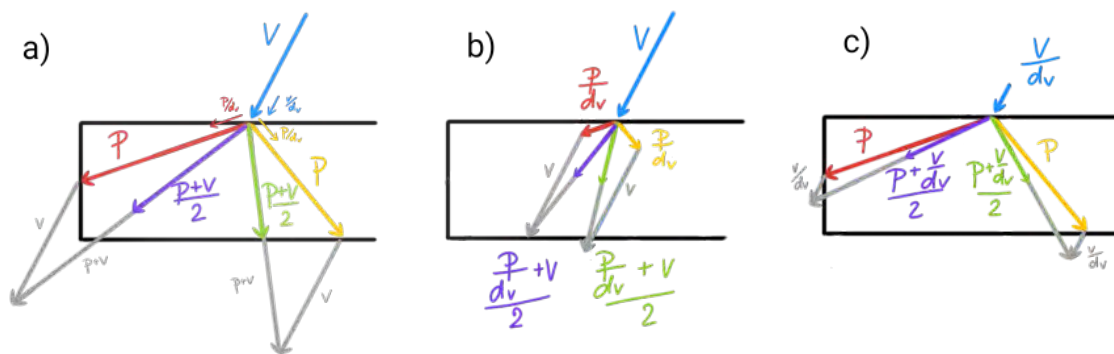
Samozrejme tento skript obsahuje implementáciu deformácie spolu s aktualizáciou meshu objektu.

DeformerInput

Komponenta **DeformerInput** je aplikovaná na všetky kladivá. Funkcionalita tohto skriptu spočíva v zaznamenaní kolíznej správy s kovovým materiálom, ktorú mu prepošle. Správu prepošle iba v prípade, ak je deformácia na danom kove povolená a sila úderu je nad určený prah pre rozpoznanie nechceného dotyku. Pri úspešnom odoslaní kolíznej správy kovovému objektu je s pomocou herného objektu **Audio Manager** spustený zvuk úderu kladiva o kov a do kontroléra odoslaný haptický impulz s pomocou funkcionality **HapticHandler**.

Deformácia objektu

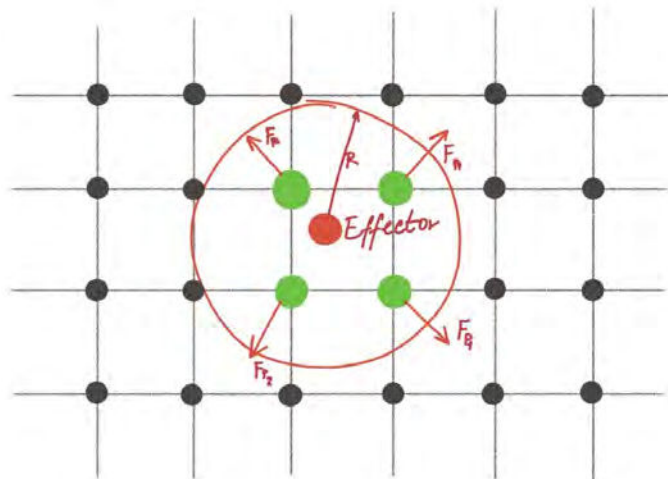
Pri deformácii kovového objektu sa sila rozkladá a ovplyvňuje celý objekt. V aplikácii je to reprezentované prechádzaním jednotlivých vertexov meshu a aplikovaním vektoru smeru a sily na vertexy, čím sa objekt deformuje. Ak by bol na všetky vrcholy použitý rovnaký vektor a sila, celý objekt by sa opticky iba posunul a deformácia by prakticky nenastala. Každý vrchol má vypočítaný posun na základe polohy v meshi a vzdialenosti od kolízneho bodu. Smer, v ktorom sa posúvajú jednotlivé vrcholy meshu, je pre každý rôzny. Výsledný vektor sa určuje spojením vstupného vektoru získaný z kolíznej udalosti vytvorenej kladivom a vektormi rozpínavosti kovu po náraze. Smer rozpínavosti sa určuje pomocou bodu kolízie kladiva s kovom a jednotlivými vrcholmi samotného kovového telesa. To s akou váhou sa vektory použijú záleží od typu deformácie. Primárna deformácia je ovplyvňovaná najviac úderom kladiva a tak je dôležitejší vstupný vektor úderu kladiva (obr. 7.9 b)). Sekundárna naopak využíva skôr vektory rozpínavosti (obr. 7.9 c)). Menej dôležitejší vektor sa utlmuje pre výpočet výsledného smeru.



Obr. 7.9: Vektory používané na zostavenie výsledného smeru deformácie (v - vstupný vektor z kolízie od kladiva, p - collision point to vertex). a) Ukážka skladania pôvodných vektorov pred aplikovaním utlmenia. b) Utlmenie rozpínavosti pre primárnu deformáciu. c) Utlmenie vstupného vektora pre sekundárnu deformáciu.

Z dôvodu, že sa sila počíta osobitne (v separátnom dočasnom poli), vektory sú normalizované a až nakoniec vynásobené výslednou silou. Sila sa počíta na základe prenosu energie po údere. To znamená, že bližšie okolie úderu sa deformuje s väčšou silou než vzdialené vrcholy na okraji. Algoritmus teda má v istej miere fungovať tak, že pri primárnej deformácii vrcholy priamo ovplyvnené úderom kladiva majú nadobudnúť tvar hlavy kladiva, avšak dôležitý faktor je tiež aktuálny stav nahratia danej časti kovového materiálu.

Deformáciu môžeme rozlíšiť na deformáciu od kladiva a deformáciu spôsobenú nákovou. Ďalej môžeme rozdeliť deformáciu od kladiva na primárnu a sekundárnu. Primárna deformácia je počítaná v mieste kolízie kovu s kladivom a blízkom okolí na ktoré priamo pôsobí úder kladiva (obr. 7.10). Takúto formu deformácie možno nazvať *subtraktívnou*, pretože vytvára priehlbiny do objektu. Sekundárna deformácia je aplikovaná na všetky ostatné časti kovu mimo hraničného polomeru primárnej deformácie. Zároveň aj z dôvodu potreby zachovania objemu kovového telesa je opačná ku primárnej deformácii, a teda má *aditívny* charakter.



Obr. 7.10: Okolie primárnej deformácie po údere kladivom v oblasti efektívneho rádiusu a výpočet deformačných vektorov od centra úderu.

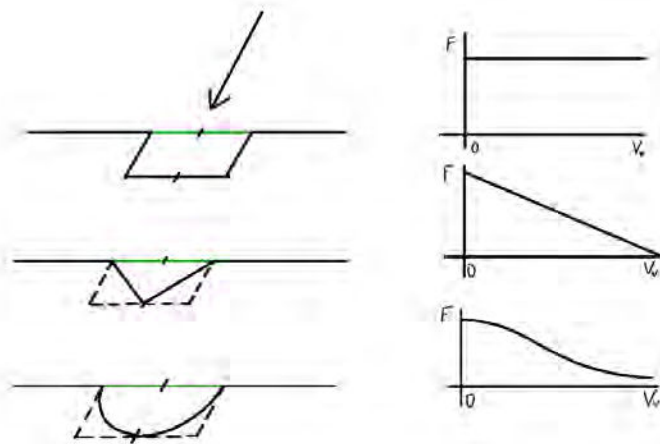
Rovnako dôležité je taktiež zachovať logiku, kedy najbližšie vrcholy meshu ku kolízii sú posúvané najväčšou silou a najvzdialenejšie najmenšou. Podobný princíp ako pri výbuchu, kedy najbližšie okolie je ovplyvnené najviac a čím vzdialenejšie miesto, tým menšia až žiadna sila nebude propagovaná.

Sila sa ďalej upravuje na základe veľkosti meshu (počtu vrcholov), keďže sa deformácia rozkladá medzi všetky vrcholy a neaplikuje sa rovnaká sila na všetky. Mesh je rozdelený na oblasť primárnej a oblasť sekundárnej deformácie. Tá istá deformačná sila je potom rozdelená medzi vrcholy jedného aj druhého typu.

Ako už bolo povedané, deformačná sila sa rozkladá medzi vrcholy meshu kovového objektu. Táto sila je braná ako výsledná rýchlosť pri kolízii kladiva s kovom. Smer a sila úderu spolu s kolíznym bodom sú použité pri celej deformácii kovového objektu.

Primárna deformácia

Keďže pri primárnej deformácii sú ovplyvňované hlavne vrcholy na základe tvaru kladiva, smeru a sily úderu, výpočet deformácie je teda závislý hlavne na týchto parametroch a menej ovplyvnený rozpínavosťou kovu (Obr. 6.18 b, c).



Obr. 7.11: Deformácia povrchu na základe kriviek grafov. Konštantná krivka vytvára veľmi hranatý zub. Lineárna krivka deformuje viac vrcholy bližšie pri kolízii a menej tie viac vzdialené. Posledná krivka upravená inverzná exponenciála má prirodzenejší tvar ku kladivám.

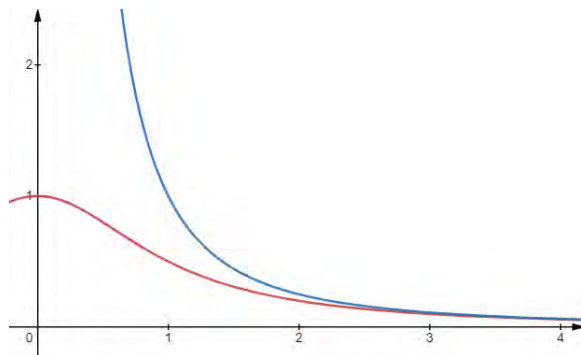
Vrcholy meshu (vertexy) patriace do primárnej deformácie sú určené na základe veľkosti a tvaru hlavy kladiva. Po náraze nie je ovplyvnený iba jeden vrchol ale viac a teda vektor smeru a sily treba rozložiť medzi ne. Ak by sa však rozdelil rovnomerne, deformácia by vytvorila zub do kovu (obr. 7.11). Kladivá využívané na deformáciu sú zaoblené tak, aby nezanechávali nechcené hrany na kovovom objekte. To platí v prípade zaoblenej hlavy využívanej na väčšie deformácie, a aj plochej hlavy využívanej skôr na vyhladzovanie deformácií. Primárna deformácia preto musí vytvoriť zaoblený tvar, najlepšie na základe hlavy kladiva. Pre zaoblenie deformácie je možné využiť *inverse-square* zákon na silu.

$$F_v = \frac{F}{d^2}$$

Táto sila sa aplikuje na vektor smeru pre posun vrcholov. Z dôvodu ošetrovania extrémov je ale možné rovnicu rozšíriť,

$$F_v = \frac{F}{1 + d^2}$$

aby viac vyhovovala požadovanej deformácii (obr. 7.12). To zaručí aby sila bola najväčšia pri vzdialenosti 0. Inak by bola najväčšia pri vzdialenosti 1 a šla by k nekonečnu čím bližšia sa k bodu priblížime.



Obr. 7.12: Základná krivka navrhnutá pre deformáciu modrá, upravená ohraničená krivka použitá pre deformáciu červená

$$v_p = \text{normalize} \left(\frac{v + \frac{e}{d_v}}{2} \right) \cdot \frac{F \cdot h}{(1 + d^2) \cdot c_p}$$

Výpočet primárnej deformácie pozostáva z dvoch častí: z výpočtu smeru a sily deformácie. Pre zloženie smeru sa používa vektor od úderu kladiva pri kolízii v a utlmený vektor rozpínavosti e pre jednotlivé vrcholy. Sila je počítaná zo vstupnej sily kolízie F upravená na základe vzdialenosti od epicentra kolízie. Keďže je sila rozložená medzi všetky ovplyvnené vrcholy (okolie kladiva, resp. jeho efektor), je potrebné ju podeliť počtom ovplyvnených vrcholov c_p . Táto sila je nakoniec ovplyvnená koeficientom deformačnej schopnosti $h = < 0; 1 >$ na základe teploty.

Sekundárna deformácia

Sekundárna deformácia je vo väčšej miere ovplyvnená rozpínavosťou kovu (Obr. 6.18 b, d). Vzorec pre výpočet je nasledovný:

$$v_s = \text{normalize} \left(\frac{\frac{v}{d_v} + e}{2} \right) \cdot \frac{F \cdot h}{(1 + d^2) \cdot c_s}$$

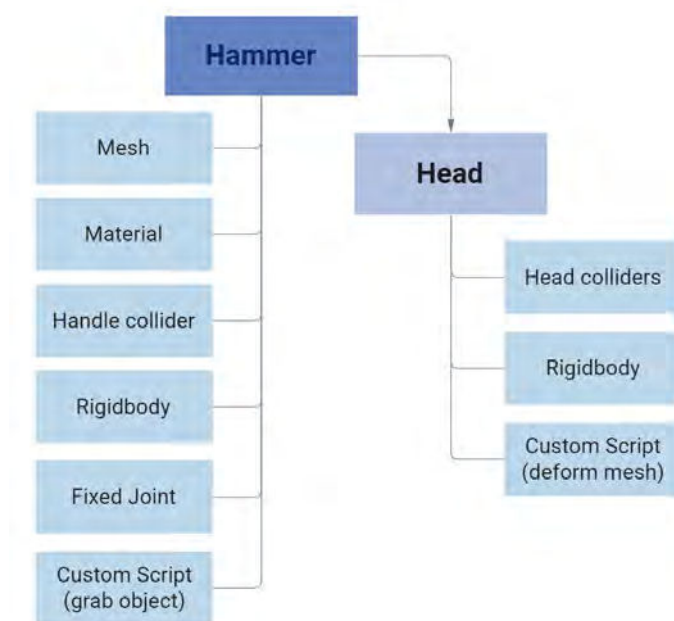
V kontraste so vzorcom pre primárnu deformáciu, výsledný normalizovaný vektor má utlmený opačný vstupný vektor, a to vektor úderu kladiva v . Hlavnú váhu má teda vektor rozpínavosti e . Sila je rozložená znovu na všetky ovplyvnené vrcholy, v tomto prípade je to však množina všetkých vrcholov, ktoré sa nenachádzali v kolíznom polomeri kladiva a teda počet vrcholov c_s je mnohonásobne väčší, čím sa docieli slabšia deformácia a nespôsobí sa tak nerealistické zväčšenie zvyšného objemu kovového objektu.

Kolízie

Deformácie kovového objektu vznikajú pomocou kladiva iba ak je tento kov položený alebo pridržený na nákove. Všetky objekty potrebujú kolízne telesá aby bolo možné zistiť kedy kolidovali a v ktorom mieste.

Pri kolíziach je nutné myslieť na to, akého typu sú objekty a aké majú nastavené vlastnosti. Kovový objekt je kinematický ak je držaný v ruke a zároveň aj ak je pri nákove. Ak by bolo kladivo z 1 kusu, tak by pri držaní bolo taktiež kinematické a tak by vznikol problém pri kolízii dvoch kinematických telies. Pri takejto kolízii nie je možné zachytiť kolíziu a teda ani kolízny bod a silu a smer úderu kladiva. Z tohoto dôvodu je nutné kladivo rozdeliť na 2 objekty, respektíve kolízne telesá hlavy kladiva (tie ktoré majú deformovať kov) sa presunú na prázdne teleso vytvorené na objekte kladiva (obr. 7.13). Taktiež na tento objekt, ktorý teraz reprezentuje deformačné kolízne telesá je potrebné pridať Rigidbody aby bolo možné zisťovať kolízne informácie. Na relatívne napojenie hlavy kladiva na rúčku a súbežný pohyb s ňou je tiež treba nastaviť **FixedJoint** medzi objektom hlavy a rúčky kladiva. **FixedJoint** zaručí, že sa hlava kladiva udrží na rúčke a nespadne na zem ovplyvnená fyzikálnymi silami.

Kvôli VR a Unity nastaveniam je tento koncept náročný na skonštruovanie...



Obr. 7.13: Zloženie kladiva z objektov a ich komponent

Kapitola 8

Záver

Práca vznikla na základe vlastného záujmu o možnosti tvorbu aplikácií vo virtuálnej realite a ich využitia. Virtuálna realita ponúka otvorené dvere možností, ktoré iba čakajú na objavenie alebo realizáciu. Umožňuje odskúšať a zažiť veci, ku ktorým by sa za celý život nebolo možné dostať. Kováčstvo patrí medzi tieto zážitkové možnosti, ku ktorým je náročné sa dostať v dnešnej dobe.

Keďže išlo o prvú prácu v engine Unity spojeného s virtuálnou realitou, bolo potrebné naštudovať, ako je možné takúto aplikáciu vytvoriť. Ďalej bolo potrebné získať informácie o technikách deformácie a vizualizácie kujných materiálov na základe historického kováčstva. S týmito znalosťami bolo možné vytvoriť návrh pre aplikáciu s demonštráciou kováčskych techník. Na základe takto vytvoreného návrhu vznikla implementácia celej aplikácie.



Obr. 8.1: Výsledná scéna

Aplikácia obsahuje základné kováčske nástroje a objekty potrebné pre deformáciu kovov. Umožňuje nahrievanie kovu rôznou rýchlosťou na základe umiestnenia v ohni a chladenie mimo ohňa buď časti kovu, alebo celého kovového objektu. Zahriaty kov je možné deformovať pomocou kladiva na nákovu. Deformácia je viditeľná v mieste úderu kladiva do kovu a zároveň sa deformačná energia rozkladá do zvyšnej časti kovového objektu. Aplikácia disponuje viacerými rozšíreniami na zlepšenie užívateľského zážitku. Medzi ne patrí oheň vytvorený pomocou časticových efektov, procedurálne materiály či už na vizualizáciu nahriateho kovu, alebo pohyblivej vody. Pridaná je tiež haptická odozva pre virtuálnu realitu pri kutí kovu alebo na upozornenie užívateľa, kedy nie je možné kov deformovať. Zároveň je implementovaná aj akustická odozva pri deformovaní kovového objektu a priestorový zvuk praskania ohňa.



Obr. 8.2: Deformácia nahriateho kovového objektu

V priebehu implementácie práce sa konalo priebežné testovanie na niekoľkých užívateľoch. Tí mali možnosť otestovať prvú iteráciu deformácie ingotu pomocou základného kladiva. Výsledkom bolo nájdenie problému, kedy pri mnohých úderoch na rovnaké miesto sa deformácia meshu nesprávala štandardne. Ďalší problém sa objavil s detekciou kolízií ingotu s nákovou a kolíziami medzi objektami držanými v rukách kvôli nastaveniam Unity a VR. V ďalšej iterácii testovania boli tieto chyby opravené a pri hre sa neobjavovali. Pridanie haptickej a akustickej odozvy bolo prijaté veľmi kladne. Podobne bolo hodnotené aj postupné nahrievanie a chladenie materiálu. Iba pri úplnom nahriatí, kedy farba kovu bola biela, bol problém s viditeľnosťou deformácie.

Výsledná aplikácia má stále niekoľko nedostatkov, ktoré by sa v budúcnosti dali odstrániť, ale z dôvodu časových a technických obmedzení nebolo možné ich vyriešiť v tejto práci. Tieto problémy sú väčšinou spôsobené limitáciami Unity pre detekciu kolízií, alebo systému pre VR, ktorý v spojení s Unity frameworkom má rôzne vedľajšie efekty pri niektorých interakciách. Vizualizácia deformácií a nahrievania by mohla mať jemnejšie a kontinuálnejšie prechody.

V budúcnosti by sa táto aplikácia mohla rozšíriť o úpravu a zrýchlenie deformácie kovu. Mohli by sa implementovať možnosti deformácie viacerými typmi kladív a na rôznych častiach nákovy, zlepšiť vizualizáciu nahriateho kovu a pridať svetelné emisie pre tento materiál. Taktiež by bolo možné pridanie schopnosti skompletovať výrobok pomocou násady a rukoväte, čo by presnejšie odpovedalo práci kováča. Aplikácia by mohla byť súčasťou väčšieho celku rôznych historických remesiel ako napríklad stolárstvo, kožiarstvo, hrnčiarstvo alebo výroba zbroje, podkov a podobne. Z výukového hľadiska by mohla byť pridaná demonštrácia kovania pomocou počítačom ovládanej postavy.

Celkový dojem z aplikácie bol vo výsledku všeobecne pozitívny. Ako hlavná kladná vlastnosť predstavovala „analogovosť“ systému kovania – odozva úderu kladiva a deformácia je

presná a úmerná jej sile. Formovaním kovového materiálu sa užívateľ nepribližuje k jednému z predvolených výsledkov, teda má absolútnu voľnosť v tvarovaní kovu. V porovnaní s inými aplikáciami s podobným účelom, na ktoré nie je nutné mať vôbec cit pre kovanie aj keď iba virtuálne.

Táto práca bola obohatením informácii o kováčstve a množstva rôznych ciest ktorými sa vydať pre jeho implementáciu. Počas jej tvorby som získala znalosti pre tvorbu VR aplikácii a možnosti implementácie deformácie materiálov.

Literatúra

- [1] *A Guide to the Types of Forging–Cold Forging and Hot Forging* [online]. [cit. 2021-01-15]. Dostupné z: <https://www.thomasnet.com/articles/custom-manufacturing-fabricating/hot-forging-cold-forging>.
- [2] *Virtual Reality Gaming PCs/ Play VR Games on PC* [online]. [cit. 2021-01-15]. Dostupné z: <https://store.hp.com/us/en/cv/omenvrpcs>.
- [3] ALQAHTANI, A. S., DAGHESTANI, D. L. F. a IBRAHIM, P. L. F. Environments and System Types of Virtual Reality Technology in STEM: A Survey. 2017.
- [4] ANGLES, B., REBAIN, D., MACKLIN, M., WYVILL, B., BARTHE, L. et al. VIPER: Volume Invariant Position-based Elastic Rods. *Proceedings of the ACM on Computer Graphics and Interactive Techniques*. júl 2019, zv. 2, č. 2, s. 1–26. DOI: 10.1145/3340260. ISSN 2577-6193. Dostupné z: <https://dl.acm.org/doi/10.1145/3340260>.
- [5] BUYUKSALIH, I., BAYBURT, S., BUYUKSALIH, G., BASKARACA, A. P., KARIM, H. et al. 3D Modelling and visualization based on the Unity game engine - Advantages and challenges. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. 2017, zv. 4, 4W4, s. 161–166.
- [6] FOUNDATION, T. B. *About* [online]. [cit. 2021-01-15]. Dostupné z: <https://www.blender.org/about/>.
- [7] HARRIES, D. a HEER, B. *Basic Blacksmithing*. Intermediate Technology Publishing, 1993. ISBN 1-85339-195-6.
- [8] KALIF, W. *Blacksmithing lesson 5:The Eight Basic Hammer Strikes* [online]. [cit. 2021-01-15]. Dostupné z: <http://www.stormthecastle.com/blacksmithing/blacksmithing-lessons/blacksmithing-lesson-5-the-8-basic-hammering-strikes.htm>.
- [9] KARASTOJKOVIĆ, Z. a BAJIĆ, N. Forge welding of biometalic axe. 2021, zv. 14, č. 2, s. 81–84.
- [10] KELLER, C. M. a KELLER, J. D. *Cognition and tool use: The blacksmith at work*. Cambridge University Press, 1996. ISBN 0-521-55239-7.
- [11] KRIŠTEK, C. P. I. *Kováčstvo - ÚLUV - Ústredie ľudovej umeleckej výroby* [online]. [cit. 2021-01-15]. Dostupné z: <http://www.uluv.sk/sk/encyklopedie/tradicne-remesla-a-domacke-vyroby/tradicne-remesla-a-domacke-vyroby/kovacstvo/>.

- [12] MANDAL, S. Brief Introduction of Virtual Reality & its Challenges. *International Journal of Scientific & Engineering Research*. apríl 2013, zv. 4, č. 4, s. 304–309. ISSN 2229-5518.
- [13] NAHRMANN, M. a MATZENMILLER, A. A critical review and assessment of different thermoviscoplastic material models for simultaneous hot/cold forging analysis. 2021, zv. 14, č. 4, s. 641–662.
- [14] RICHARDSON, M. T. *Practical Blacksmithing: The Four Classic Volumes in One*. Chartwell Books, 2017. ISBN 978-0-7858-3539-4.
- [15] SIMONDS, B. *Blender Master Class: A Hands-on Guide to Modeling, Sculpting, Materials and Rendering*. No Starch Press, 2013. ISBN 978-1-59327-477-1.
- [16] TAYLOR, J. *Modeling vs. Sculpting: How do you know which to use* [online]. 2016 [cit. 2021-01-15]. Dostupné z: <https://www.methodj.com/modeling-vs-sculpting/>.
- [17] TEAM, B. D. *Introduction - Blender Manual* [online]. [cit. 2021-01-15]. Dostupné z: https://docs.blender.org/manual/en/latest/sculpt_paint/texture_paint/introduction.html.
- [18] TECHNOLOGIES, U. *Unity - Manual: 3D Physics for object-oriented objects* [online]. 2020 [cit. 2021-01-15]. Dostupné z: <https://docs.unity3d.com/2020.2/Documentation/Manual/PhysicsOverview.html>.
- [19] TECHNOLOGIES, U. *Unity - Manual: Graphics* [online]. 2020 [cit. 2021-01-15]. Dostupné z: <https://docs.unity3d.com/2020.1/Documentation/Manual/Graphics.html>.
- [20] TECHNOLOGIES, U. *Unity - Manual: Unity User Manual 2020.2*. [online]. 2020 [cit. 2021-01-15]. Dostupné z: <https://docs.unity.cn/2020.2/Documentation/Manual/UnityManual.html>.
- [21] TECHNOLOGIES, U. *Unity - Manual: XR* [online]. 2020 [cit. 2021-01-15]. Dostupné z: <https://docs.unity3d.com/2020.1/Documentation/Manual/XR.html>.
- [22] TECHNOLOGIES, U. *Unity engine visual scripting* [online]. 2020 [cit. 2021-01-15]. Dostupné z: <https://unity.com/products/unity-visual-scripting>.
- [23] XIE, J. Research on key technologies base Unity3D game engine. 2012, s. 695–699.

Kapitola 9

Obsah priloženej SD karty

Ku práci je priložená SD karta s nasledujúcim obsahom:

- VR/ priečinok obsahujúci Unity projekt verzie 2021.2.7
 - Assets/ priečinok s dátami potrebnými pre beh aplikácie
 - Scripts/ vlastný zdrojový kód
 - Objects/ 3D modely objektov vytvorených v softwari Blender
 - Materials/ Materiály, textúry pre modely, shadery a skybox
 - Scenes/ Metadáta objektu scény
 - Audio/ Zvukové stopy použité v aplikácii
 - ostatné priečinky obsahujúce konfiguráciu vývojárskeho prostredia Unity a metadáta potrebné pre správny preklad a spustenie projektu
- App/ priečinok obsahujúci preloženú aplikáciu
- vr_blacksmith.pdf text technickej správy
- Doc/ zdrojové súbory technickej správy
- vr_blacksmith.mp4 video