

ZÁPADOČESKÁ UNIVERZITA V PLZNI  
**FAKULTA STROJNÍ**

Studijní program: B2341 Strojírenství  
Studijní zaměření: Informační a komunikační technologie  
ve strojírenském podniku

**BAKALÁŘSKÁ PRÁCE**

Možnosti aplikace virtuálních lidí ve vybraném nástroji

Autor: **Petr KLEČANSKÝ**  
Vedoucí práce: **Ing. Petr HOŘEJŠÍ, Ph.D.**

Akademický rok 20012/2013

## **Prohlášení o autorství**

Předkládám tímto k posouzení a obhajobě bakalářskou/diplomovou práci, zpracovanou na závěr studia na Fakultě strojní Západočeské univerzity v Plzni.

Prohlašuji, že jsem tuto bakalářskou/diplomovou práci vypracoval samostatně, s použitím odborné literatury a pramenů, uvedených v seznamu, který je součástí této bakalářské/diplomové práce.

V Plzni dne: .....

.....  
podpis autora

## ANOTAČNÍ LIST DIPLOMOVÉ (BAKALÁŘSKÉ) PRÁCE

<b>AUTOR</b>	<b>Příjmení</b> Klečanský	<b>Jméno</b> Petr	
<b>STUDIJNÍ OBOR</b>	2341R001 Informační a komunikační technologie ve strojírenském podniku		
<b>VEDOUCÍ PRÁCE</b>	<b>Příjmení (včetně titulů)</b> Ing. Hořejší, Ph.D.	<b>Jméno</b> Petr	
<b>PRACOVISŤE</b>	ZČU - FST - KPV		
<b>DRUH PRÁCE</b>	<b>DIPLOMOVÁ</b>	<b>BAKALÁŘSKÁ</b>	<b>Nehodící se škrtněte</b>
<b>NÁZEV PRÁCE</b>	Možnosti aplikace virtuálních lidí ve vybraném nástroji		

<b>FAKULTA</b>	strojní	<b>KATEDRA</b>	KPV	<b>ROK ODEVZD.</b>	2013
----------------	---------	----------------	-----	--------------------	------

### POČET STRAN (A4 a ekvivalentů A4)

<b>CELKEM</b>	54	<b>TEXTOVÁ ČÁST</b>	42	<b>GRAFICKÁ ČÁST</b>	12
---------------	----	---------------------	----	----------------------	----

<b>STRUČNÝ POPIS (MAX 10 ŘÁDEK)</b> <b>ZAMĚŘENÍ, TÉMA, CÍL POZNATKY A PŘÍNOSY</b>	Cílem bakalářské práce je popsat tvorbu choreografických scén v SW Face Poser, který je součástí balíčku Source SDK, a implementovat virtuální postavy do existujícího virtuálního výrobního prostředí. Vytvořené návody rozšiřují výukové podklady kurzu KPV/DPVR – digitální podnik a virtuální realita.
<b>KLÍČOVÁ SLOVA</b> <b>ZPRAVIDLA JEDNOSLOVNÉ POJMY, KTERÉ VYSTIHUJÍ PODSTATU PRÁCE</b>	digitální továrna, skeletální animace, virtual human, Source SDK, Valve Face Poser, Valve Hammer Editor

## SUMMARY OF DIPLOMA (BACHELOR) SHEET

<b>AUTHOR</b>	<b>Surname</b> Klečanský	<b>Name</b> Petr	
<b>FIELD OF STUDY</b>	2341R001 Information and Communication Technology in Industrial Management		
<b>SUPERVISOR</b>	<b>Surname (Inclusive of Degrees)</b> Ing. Hořejší, Ph.D.	<b>Name</b> Petr	
<b>INSTITUTION</b>	ZČU - FST - KPV		
<b>TYPE OF WORK</b>	<b>DIPLOMA</b>	<b>BACHELOR</b>	<b>Delete when not applicable</b>
<b>TITLE OF THE WORK</b>	Application possibilities of virtual humans in selected tool		

<b>FACULTY</b>	Mechanical Engineering	<b>DEPARTMENT</b>	Industrial Engineering and Management	<b>SUBMITTED IN</b>	2013
----------------	------------------------	-------------------	---------------------------------------	---------------------	------

### NUMBER OF PAGES (A4 and eq. A4)

<b>TOTALLY</b>	54	<b>TEXT PART</b>	42	<b>GRAPHICAL PART</b>	12
----------------	----	------------------	----	-----------------------	----

<b>BRIEF DESCRIPTION</b> <b>TOPIC, GOAL, RESULTS AND CONTRIBUTIONS</b>	The goal of the bachelor's thesis is to describe the creation of choreographed scenes in SW Face Poser, which is the part of Source SDK package, and implement virtual characters to an existing virtual manufacturing environment. Created guides extend educational materials for course KPV / DPVR – Digital Factory and Virtual Reality.
<b>KEY WORDS</b>	digital factory, skeletal animation, virtual human, Source SDK, Valve Face Poser, Valve Hammer Editor

## Obsah

1	Úvod.....	8
2	Virtuální lidé .....	9
2.1	Skeletal Animation .....	9
2.1.1	Skinning .....	10
2.1.2	Inverzní kinematika.....	11
2.2	Motion Capture.....	12
2.2.1	Mechanické systémy .....	12
2.2.2	Optické systémy .....	12
2.2.3	Elektro-magnetické systémy .....	13
2.2.4	Zpracování MoCap dat.....	14
2.3	Motion capture vs ruční animace.....	14
3	Analýza současného stavu.....	15
3.1	Moderátoři .....	15
3.2	Postavy ve videohrách .....	16
3.3	Aplikace v průmyslu.....	17
3.4	Aplikace ve zdravotnictví.....	19
3.5	Aplikace v armádě .....	20
3.6	Autonomous virtual characters .....	21
4	Popis vybraného nástroje .....	22
4.1	Source SDK .....	22
4.2	Adresářová struktura.....	23
4.3	Face Poser.....	23
5	Návody pro práci ve Face Poseru.....	25
5.1	Vytvoření rámce scény .....	25
5.2	Načtení modelu postavy .....	25
5.3	Příprava a vložení zvukové stopy .....	26
5.3.1	Phoneme Tool .....	27
5.4	Vkládání a úprava gest .....	28
5.4.1	Ramp Tool.....	30
5.5	Animace obličeje .....	30
5.6	Implementace scény pomocí VHE .....	31
5.6.1	Odezva na podnět .....	32
5.7	Orientace a pohyb v prostoru.....	33
5.7.1	Look at actor.....	33

5.7.2	Face to actor .....	33
5.7.3	Move to actor .....	34
5.8	Interakce .....	35
5.8.1	Section Pause.....	35
5.8.2	Loop .....	36
5.8.3	Sub-scene .....	37
5.8.4	Fire Trigger .....	37
6	Komplexní ukázka.....	38
6.1	Popis ukázky.....	38
6.1.1	Úvodní scéna .....	38
6.1.2	Prezentační místnost.....	38
6.1.3	Výrobní hala .....	40
6.1.4	První patro .....	40
6.1.5	Vývojové oddělení .....	41
6.2	Zhodnocení .....	41
6.3	Storyboard .....	42
7	Tvorba výukového e-booku .....	43
8	Závěr.....	44
	Zdroje .....	45
	Příloha č. 1.....	47
	Skript game_sounds_manifest.txt .....	47
	Příloha č. 2.....	53
	Skript level_sounds_konstrukce.txt .....	53

## **Použité zkratky**

VR – Virtuální Realita

VH – Virtual Human

DF – Digital Factory

LBS – Linear Blend Skinning

IK – Inverzní Kinematika

AI – Artificial Intelligence

PLM – Product Lifecycle Management

AVC - Autonomous Virtual Character

KPV – Katedra průmyslového inženýrství a managementu

DPVR – Digitální podnik a virtuální realita

SDK – Software Developers's Kit

VHE – Valve Hammer Editor

NPC – Non Player Character

VCD – Valve Choreography Data

BSP – Binary Space Partitioning

VTF – Valve Texture File

VMT – Valve Material Type

SMD – Valve Studiomdl Data File

GFC – Game Cache File

WAV – Waveform Audio File

RC – Right Click

LC – Left Click

ADPCM – Adaptive Delta Pulse Code Modulation

PCM – Pulse Code Modulation

LCS – Logic Choreographed Scene

MDL – Half-Life Model File

# 1 Úvod

Když se v druhé polovině dvacátého století začal používat termín virtuální realita (VR), přinesl s sebou velká očekávání. Hlavní myšlenka byla, že tato technologie bude schopna vytvářet imaginární světy nerozeznatelné od skutečných. Technologie však ještě nebyla připravena dělat počítačem generované světy zcela věrohodně. V dnešní době se tato myšlenka přehodnotila a VR je spíše o vytváření přijatelných reprodukcí reálných objektů či prostředí za účelem simulace, tréninku, designu a v neposlední řadě zábavy. [1]

Virtuální realita byla ve svých počátcích omezena hlavně nedostatečným výkonem hardwaru počítačů. S rychlým rozvojem počítačových technologií se hranice možností stále rozšiřují, limitující je pořizovací cena zařízení a softwarového vybavení. VR je možno rozdělit do tří základních kategorií. Pasivní VR, která je v dnešní době používána především ve filmovém průmyslu. Aktivní VR, ve které je již možné prostředí aktivně zkoumat bez možnosti ovlivnit prostředí. A do třetice aplikace s interaktivní VR, ve kterých lze prostředí aktivně ovlivňovat.

Dnešní nejmodernější technologie nabízí řadu nástrojů, které umožňují poměrně realisticky zrekonstruovat modely skutečného světa ve virtuálním prostředí ve třech rozměrech. Virtuální prostředí jsou obecně složena ze statických a dynamických objektů a mohou zahrnovat 3D grafické objekty, 3D zvuky, obrázky a videa. Problémem většiny z nich je, že mohou nabídnout jen možnost pasivního prozkoumání scény a minimum interakce. Přitom aktivní účast a zapojení uživatele vyžaduje interakci a komunikaci mezi uživatelem a virtuálním okolím. Klíčem k tomuto zapojení mohou být virtuální lidé. [2]

Ponoření (immersion) se do děje a vytvoření dojmu přítomnosti (presence) uvnitř VR umožňují technické prostředky, jako jsou virtuální brýle nebo celé projekční sály pro stereoskopickou projekci, prostorové reproduktory, speciální datové rukavice se zpětnou vazbou a snímače polohy či pohybů (trackovací zařízení).

Cílem teoretické části této práce je zmapovat použití virtuálních lidí (Virtual Humans - VH) ve virtuálním prostředí se zaměřením na současný stav v tomto oboru. V praktické části pak bude následovat popis konkrétního softwarového nástroje pro implementaci VH do výrobního prostředí a tvorba návodů k této aplikaci. Virtuální lidé mohou významně oživit návrhy pracovišť a výrobních systémů v konceptu Digitálního podniku.

Digitální podnik (Digital Factory - DF) je termín, který se stále více dostává do povědomí vedoucích a technických pracovníků ve výrobních podnicích. V obecném pojetí pokrývá tato strategie celou oblast přípravy výroby a slouží především jako výchozí myšlenkový směr k digitalizaci plánování, simulace a optimalizace. Systémy digitální továrny nacházejí uplatnění zejména v automobilovém, leteckém a lodním průmyslu. Představují další logický krok v postupném vytváření specializovaných nástrojů pro podporu procesů v celém životním cyklu výrobku. Již během fáze plánování mohou být ověřeny všechny části výrobního systému takovým způsobem, že následná reálná výroba pak bude zajištěna z hlediska kvality, času a nákladů. [3][4]

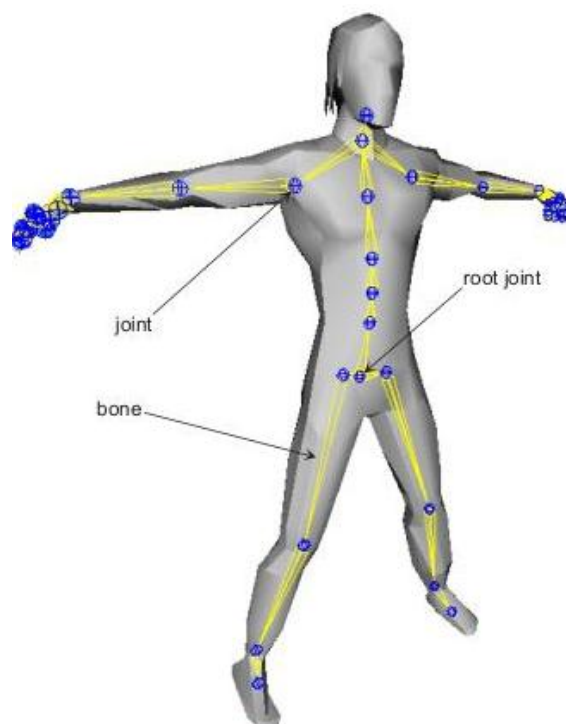


## 2 Virtuální lidé

Virtuální lidé, někdy též nazývaní „avataři,“ jsou počítačem vytvořené postavy s lidskou podobou. Tato lidská forma je důležitá, a jak se zdá, je schopna zvláštním způsobem působit na uživatele. To znamená, že uživatel reaguje na virtuálního člověka velmi přesvědčivě, odlišně než např. na virtuální auta, letadla nebo budovy. Tito virtuální lidé jsou rozhodující pro aplikace prezentující obydlené světy a partneři pro uživatelskou interakci. Virtuální lidé, stejně jako jejich lidské protějšky, jsou nesmírně komplexní bytosti. Proto je také velmi obtížné je animovat, zautomatizovat či s nimi komunikovat. Ve všech těchto oblastech musí dojít k významnému pokroku pro realizaci přesvědčivých virtuálních lidí. [5] Často diskutovaný je problém tzv. „uncanny valley“, tedy záhadného údolí. Podle této hypotézy může i poměrně zdařilá virtuální postava, která se podobá člověku, vyvolat negativní reakci pozorovatele, i když se bude pohybovat a vypadat téměř přirozeně. Vyřešení tohoto problému označil James Cameron jako nejtěžší úkol při práci na filmu Avatar. V této kapitole následuje popis vytváření virtuálních postav a animačních metod.

### 2.1 Skeletal Animation

Skeletální animace je technika počítačové animace, ve které je model reprezentován povrchem (skin, mesh) a kostrou (skeleton, rig). Kostra je složena z hierarchické struktury kloubů (uzlů) vzájemně spojených kostmi (hranami) a slouží k animaci povrchu, který je použit k vykreslení postavy. Každý kloub může rotovat kolem libovolné osy a ovlivňuje všechny potomky. Například rotace ramene otáčí celou paži. Skelet i povrchový model se navrhuje v referenční poloze, což je typicky stojící postava s rozpaženými pažemi. Pomocí této, pro animaci virtuálních lidí standardně používané metody, se animační proces stává logičtější a intuitivnější, navíc je oproti per-vertex animaci (animace každého vrcholu zvlášť) i méně náročná na paměť. [6][7] Avšak při animaci obličejů, tedy mimiky, se skeletální animace nevyužívá.



Obr. 2.1 . ukázka Skeleton [O1]

### 2.1.1 Skinning

Skinning je proces, při kterém se vytváří spojení mezi skeletem a povrchem animovaného modelu. Existují různé přístupy, jak tohoto spojení dosáhnout. Tyto přístupy můžeme zhruba rozdělit do tří hlavních kategorií: skeleton-based deformations, data-driven methods a physics-based approaches. [1]

#### Skeleton-based deformations

Základem této metody je, že každému vrcholu povrchu se přiřadí jeden nebo více řídících kloubů s rozdílnými vahami. Deformace vrcholů je poté automaticky vypočtena. Přestože tato metoda dává rychlé výsledky a není příliš náročná na paměť, jejími nedostatky jsou nežádoucí deformace v případě příliš velké změny úhlu kloubů.

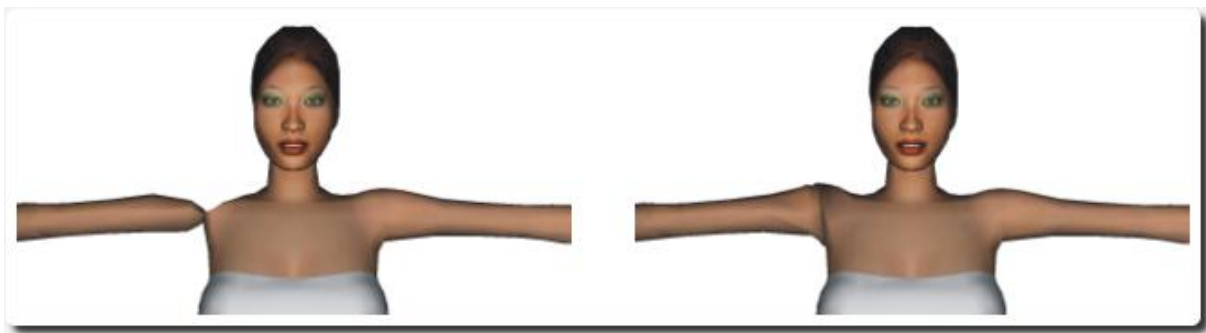
#### Data-driven methods

Data-driven methods, tedy v překladu „daty řízené metody“, se snaží překonat limity geometrické deformace povrchu pomocí záznamů z různých pozic a jejich „míchání“ během animace. Výsledného efektu realistického tvaru je dosaženo pomocí snímání reálných lidí. Tyto metody se začaly rozvíjet s příchodem 3D skenerů a trackovacích zařízení. Tato zařízení nám umožňují získat spolehlivé a přesné údaje o tak komplexních tvarech, jako jsou deformace povrchu lidského těla. Základní myšlenkou je, že je možné zevšeobecnit údaje získané ze sady jednotlivců např. pomocí statistické analýzy.

#### Physics-based approaches

Physics-based approaches, tedy doslovně přeloženo „přístupy založené na fyzice“ byly navrženy za účelem zavedení více realistických deformací. Pomocí algoritmů, ve kterých je možno zohlednit vnější (gravitace, vítr, ...) i vnitřní síly (síla ve svalech), se provede výpočet dynamiky pohybu.

Další možné rozdělení skinningu je dle počtu přiřazených kloubů. Jednodušší přístup tzv. rigid skinning přiřazuje každému vrcholu „meshe“ právě jeden kloub, což je dostačující pro vrcholy dále od kloubů. Při větších rotacích se na vrcholech poblíž kloubů objevují nepřirozené deformace. Tento problém se však vyskytuje i u druhého přístupu, tzv. linear blend skinning (LBS), kde je každý vrchol ovlivňován několika klouby. Tato metoda dává hladké deformace pokožky a je podporována i grafickým hardwarem. LBS je často používána při vývoji PC her, díky jednoduché implementaci a velmi rychlým výpočtům. [7]



Obr. 2.2 - ukázka Linear Blend Skinning [O2]

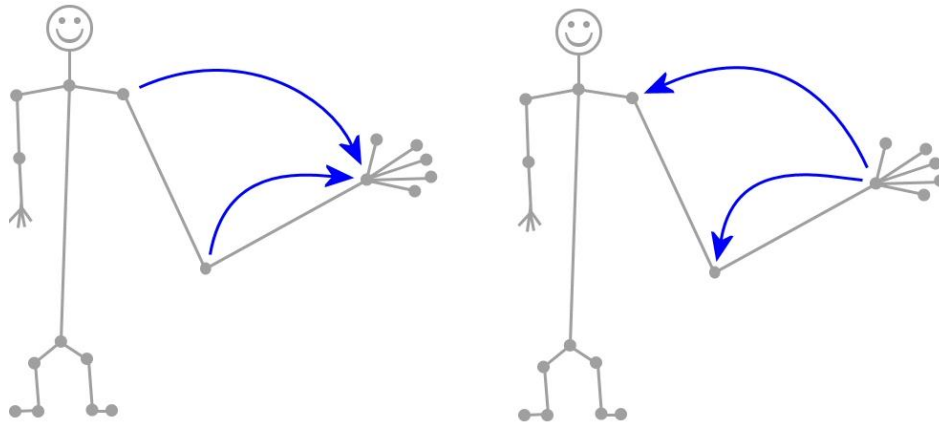
Povrch modelu je možné tvořit „ručně“, ale pro dosažení více realistických humanoidů je výhodnější použít technologii zvanou **3D body scanning**. Pomocí těchto sofistikovaných zařízení lze jednoduše a velice rychle získat realistický digitální model reálné osoby. Na nejnovějších scannerech lze dosáhnout přesnosti v řádu desetin milimetru. Scanování funguje na principu snímání modelu ze všech stran a následné rekonstrukci pomocí SW.



Obr. 2.3 - ukázka 3D body scanning [O3]

### 2.1.2 Inverzní kinematika

Pokrok v oblasti výzkumu skeletální animace přináší další výhody. Jednou z nich je použití inverzní kinematiky (IK), což umožňuje ovládat postavu nepřímo jako kinematický řetězec. Pomocí kinematických rovnic se definují vztahy mezi úhly kloubů a pozicí (postojem) modelu. Pro požadovanou pozici modelu či jeho části se poté vypočtou úhly kloubů. Tento postup je pro animátory výhodnější, protože mohou definovat prostorovou konfiguraci postavy pomocí jejích částí a nemusí složitě manipulovat s úhly kloubů jako při kinematice dopředné. Pro představu, animátor přesune ruku modelu do požadované polohy (např. uchopení předmětu) a algoritmus dopočte správné úhly zápěstí, lokte a ramene. Pomocí této metody se např. efektivně dosáhne, aby se postava při animaci chůze přesně dotýkala země a nechodila ve vzduchu či naopak pod povrchem. Kontrolní body, které jsou připojeny na všechna důležitá místa řetězce, mohou být řízeny manuálně nebo efektivněji pomocí externích dat, získaných pomocí motion capture (kapitola 2.2). [6][8] Na obrázku 2.4 je vidět rozdíl mezi dopřednou (vlevo) a inverzní kinematikou (vpravo).



Obr. 2.4 - ukázka FK a IK [O4]

## 2.2 Motion Capture

Motion capture, zkráceně MoCap, přeloženo do češtiny „snímání pohybu“, je termín, kterým se označuje proces nahrávání pohybu skutečného objektu a jeho převedení na digitální model. [9] Tato metoda je dnes neodmyslitelně spjatá s animací 3D postav, protože poskytuje informace o pohybu končetin, držení těla, gestech, mimice atd. Technologie pro snímání pohybu se dají, dle použité technologie, zhruba rozdělit do tří kategorií – mechanické, optické a magnetické. Pro potřeby animace VH se ale používají v největším měřítku systémy optické.

### 2.2.1 Mechanické systémy

V případě mechanických systémů se herec oblékne do speciálního obleku (exoskeleton), kde je každý kloub připojen k úhlovému kodéru. Pohyb každého snímače na obleku je zaznamenán v počítači, který tím, že zná relativní polohu snímače, pomocí SW zrekonstruuje pohyb na obrazovce. [10]

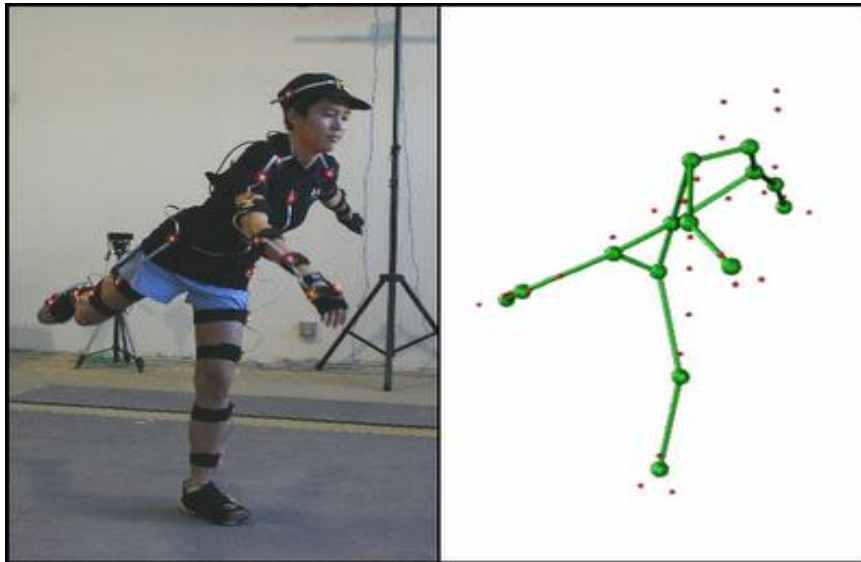


Obr. 2.5 - ukázka Mechanical MoCap [O5]

### 2.2.2 Optické systémy

Metoda, při které je na všechny důležité části sledovaného objektu připevněno množství speciálních značek (markerů), ty jsou snímány pomocí prostorově umístěných kamer. Systém je pak schopen v reálném čase zjistit polohu každého markeru, čímž se získá informace o umístění klíčových bodů v animovaném modelu. Výhodou těchto optických systémů je, že jsou

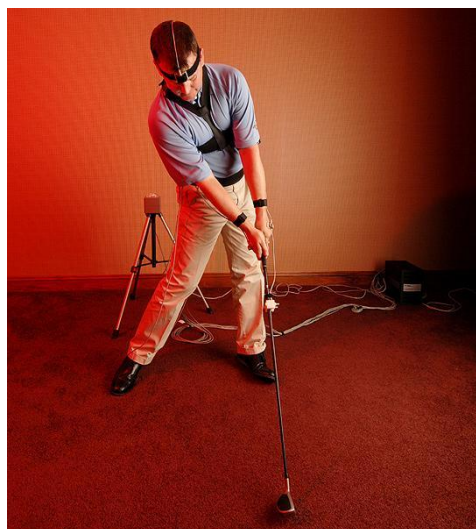
velice přesné a flexibilní. Markery jsou obvykle kuličky o průměru 2 – 3 mm, což umožňuje jejich upevnění téměř kamkoliv. Rozlišujeme aktivní a pasivní optické systémy. V pasivních optických systémech markery světlo pouze odrážejí, zatím co v aktivních systémech jsou přímo napájené a světlo emitují. [9] Mezi nevýhody patří např. citlivost senzorů na světlo, náročná kalibrace a v neposlední řadě vysoké pořizovací náklady. Velkým problémem může být tzv. „Line of Sight“, tedy že marker musí na senzor „vidět“.



Obr. 2.6 - ukázka Optical MoCap [O6]

### 2.2.3 Elektro-magnetické systémy

V této technologii se využívá snímání magnetického pole senzoru k zaznamenání pohybu. Na tělo herce jsou upevněny kabelové magnetické senzory, které jsou schopné zachytit pozici i orientaci. Každý senzor má svou jedinečnou identifikaci v systému. Oproti optickým systémům je výhodou, že nemůže dojít k zastínění senzoru. Na druhou stranu je tato metoda o poznání méně přesná, senzory jsou citlivé na kov a herce omezují v pohybu připojené kabely. [2]



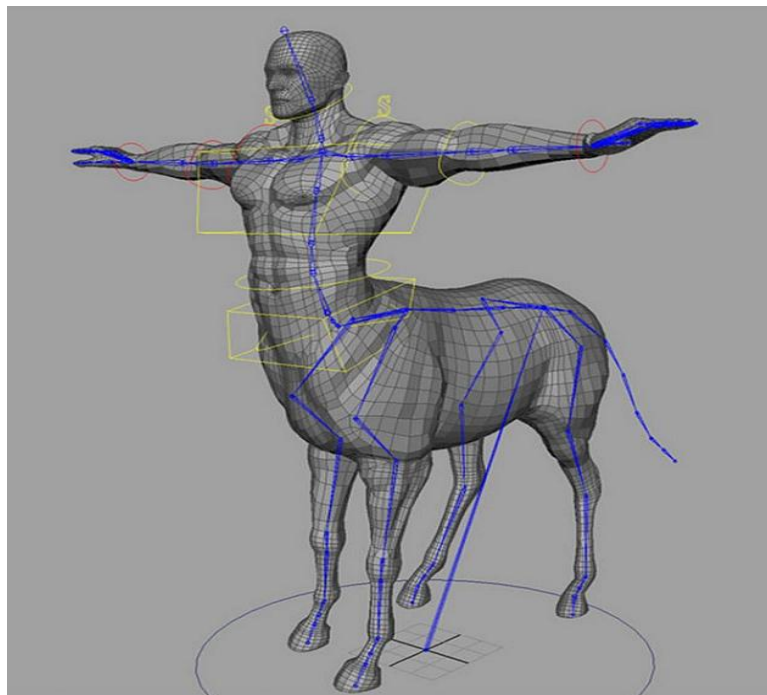
Obr. 2.7 - ukázka magnetic mocap [O7]



#### 2.2.4 Zpracování MoCap dat

Po dokončení snímání pohybu probíhá automatické zpracování dat pomocí SW. Tím však práce ještě nekončí, v další fázi se musí nasnímaná animace tzv. „namapovat“ na animovaný model. Tomuto procesu se říká „labeling“ a spočívá vlastně v přiřazení nasnímaných trajektorií jednotlivým kontrolním bodům definovaným na skeletonu modelu. [11] Když animátor potřebuje, aby se různé virtuální postavy pohybovaly podobně, může využít tzv. „retargeting“, což lze přeložit jako přesměrování animace. Toto přesměrování umožňuje využít data získaná pomocí MoCap i pro animaci zcela odlišných kinematických struktur.

Všechny metody popsané v kapitolách 2.1 a 2.2 lze úspěšně aplikovat nejen při animaci VH, ale např. i při tvorbě zvířat či libovolných virtuálních bytostí.



Obr. 2.8 - ukázka Centaur Concept [O8]

### 2.3 Motion capture vs ruční animace

Každá z výše popisovaných metod má své klady a zápory. Motion capture byl donedávna považován za velice kontroverzní nástroj, protože vyčištění dat ze záznamu bylo kolikrát stejně časově náročné jako ruční animace. Výrobci zaznamenávacích systémů si toho byli velice dobře vědomi, a tak se pomocí nutného pokroku v SW podpoře této technologie dosáhlo použitelnosti a efektivnosti MoCap při generování animací.

Využitím MoCap lze tedy dosáhnout značných časových i nákladových úspor, které by jinak byly vynaloženy na práci animátora. [11] Vše je ale na druhé straně vyváжено nutnou investicí do pořízení či pronájmu zařízení. Jednou z nevýhod použití digitálního zaznamenávání pohybů jsou bezesporu anatomické limity člověka. Pomocí motion capture tedy získáme zcela jistě přirozenější animaci, avšak ruční animace je doposud nezbytnou součástí práce animátorského týmu.

### 3 Analýza současného stavu

Možnosti využití virtuálních lidí jsou téměř neomezené a s pokrokem techniky se stále rozrůstají. Své uplatnění najdou v řadě oborů od zábavy až po armádní aplikace. V poslední době jsou stále více populární samostatně jednající postavy. V oblasti speciálních efektů jsou virtuální postavy použity k nahrazení herců při scénách, které jsou příliš obtížné či drahé na výrobu se skutečnými herci. V počítačových hrách je najdeme jako herce řízené scénářem, ale schopné reagovat na libovolné podněty uživatele a přizpůsobit své chování. V oblasti simulace umožňují postavy studovat nové situace a jejich možné důsledky. [12] Dále si uvedeme konkrétní příklady nasazení virtuálních lidí.

#### 3.1 Moderátoři

S nástupem 3D grafiky je nyní možné vytvořit postavy, které se mohou hýbat a mluvit. S rozvojem technologií jako je např. rozpoznání či syntéza řeči je možné tyto bytosti nejen pozorovat, ale i s nimi aktivně komunikovat. Společnost Matrox Graphics Inc. vytvořila Matrox Virtual Presenter pro Microsoft PowerPoint. Je to doplňkový modul pro tento široce využívaný software, který používá vysoce realistické mluvící hlavy jako inovativní a efektivní prostředek k elektronickému představení prezentace v angličtině. [2]

Dalším příkladem je zřejmě první virtuální moderátorka zpráv Ananova, vytvořená společností Ananova a jejich partnery. Designéři strávili mnoho času návrhem různých kombinací obličejových rysů, aby mohli vytvořit atraktivní a nenucený vzhled. Největší výzvou vývojářů bylo, jak vytvořit plně animovanou postavu, která by byla schopná přizpůsobit svůj výraz a mimiku specifickému obsahu zpráv. Tato služba sloužila ke zprostředkování novinek pro webové uživatele. Pro vytvoření hlasu byl použit software od Lernout&Hauspie, který je schopen převést text do řeči. Tento SW je schopen zajistit i synchronizaci řeči a úst, tzv. „lip-syncing“. [2] Od roku 2004 již tato služba není dostupná.



Obr. 3.1 - ukázka Ananova [09]

V poslední době je také stále více webových stránek schopno implementovat virtuální postavy pro zlepšení prezentace či zábavnosti webu. Sims a Carol Wideman vyvinuli za pomoci National Science Foundation software SigningAvatar. Tento SW umožňuje zobrazovat na strán-

kách 3D animaci postavy, která používá znakovou řeč a pomáhá tak sluchově postiženým dětem rozvíjet jazyk a čtení. Tato technologie má podporu učitelů i počítačových odborníků a používá se na několika školách v Americe. [2]

### 3.2 Postavy ve videohrách

Video hry obsahující VH jsou dnes naprosto běžné. Mnoho her používá výpočetní výkon počítače k vytvoření virtuálních světů a postav, se kterými může „hráč“ dynamicky spolupracovat. Nejrozšířenější technikou, kterou tvůrci her využívají k animaci postav, je Motion Capture (kapitola 2.2). To znamená, že pohyby skutečného člověka jsou zachyceny na počítači a následně využity k realističtějšímu pohybu postavy ve hře. Čím přirozenější výsledek chceme, tím více mezikroků každého pohybu musíme zaznamenat. Např. kopnutí v bojovém simulátoru se skládá z pohybu kyčle, stehna, kolena, lýtka a nakonec kotníku a chodidla. Každý z jednotlivých pohybů se skládá z několika snímků, aby se zamezilo trhané animaci. [2]

Jedním z nejrozšířenějších animačních a vizualizačních programů je Autodesk 3ds Max, který je určen pro tvorbu vizuálních efektů, animaci postav a tvorbu počítačových her. Jen pro představu, tímto programem byly vytvořeny hry Unreal Tournament, Grand Theft Auto: Vice City Warcraft III, Simcity, Tomb Raider, Spider-Man a mnoho dalších. [13]

Nejúspěšnější hrdinkou videoher je, i podle Guinnessovy knihy, postava Lara Croft ze série Tomb Raider od společnosti Eidos Interactive. Lara Croft představuje atraktivní archeoložku ve stylu Indiana Jonese, která se snaží pomocí rozumu a zbraní porazit nepřátele a získat magický artefakt. Vytvoření všech animací, jako např. lezení, střelení, plavání, kotouly, kopaní, je velice složitý, časově náročný a komplexní proces. Při vývoji hry Tomb Raider: Underworld sloužila jako předloha pro zachycení pohybů britská gymnastka Alison Carroll. Výsledný pohyb postavy působí opravdu poměrně realisticky. Na obrázku 3.2 je vidět reálná i virtuální postava.



Obr. 3.2 . ukázka Tomb Raider Underworld [O10]





Obr. 3.3 - ukázka The Sims 3 [O11]

Další velice úspěšnou herní sérií je The Sims od společnosti Maxis Software, jejímž principem je vytváření a kontrola virtuálních lidí. Hra se zaměřuje na řízení každodenních činností tzv. „Simíků“ jako je spaní, jídlo, vaření, koupání, oblékání atd. Presentace umělé inteligence (AI - Artificial Intelligence) je velice zdařilá. Postavy reagují na vnější podmínky samostatně, i když je někdy nutný zásah hráče. Tvůrce hry Will Wright řekl: „*V podstatě chci, aby mě hra, co programuji, překvapila. Vidím, jak se dějí legrační věci, které nebyly explicitně naprogramované, ale jsou jen funkcí jednoduchých interakčních pravidel. A to je pro mne skutečné jádro umělého života.*“ AI má dopad na hry vytvářením entit, které se chovají jako reální lidé se svými vlastnostmi, schopnostmi a zkušenostmi. [13]

### 3.3 Aplikace v průmyslu

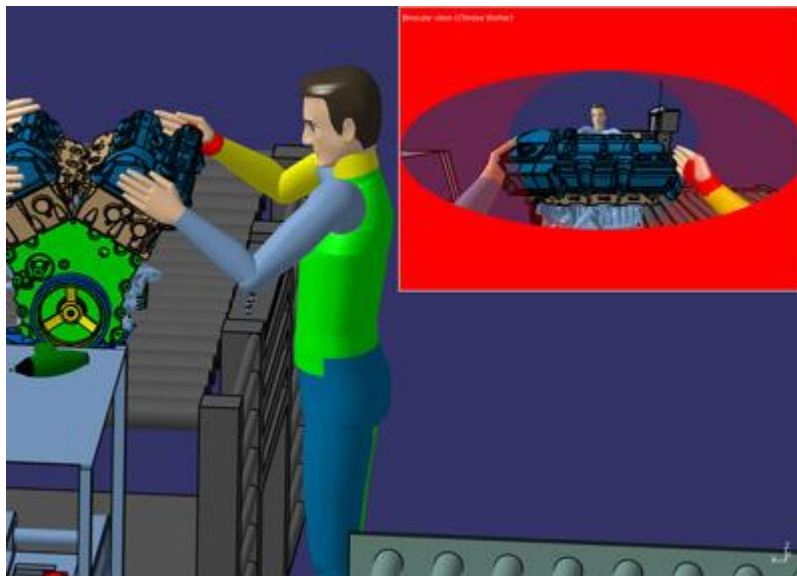
Existuje celá řada možností, jak využít VH v průmyslovém podniku. Například provádění simulací lidského těla ve výrobním prostředí a sledování faktorů, které na pracovníky působí. Specializované nástroje slouží k pochopení a optimalizaci vztahů mezi lidským tělem, výrobky a stroji. Pomocí těchto nástrojů lze lidské tělo nastavit do pracovní polohy a sledovat statické i dynamické zatížení nebo zjistit, zda a za jakých podmínek je daný výrobek lidmi vyrobitelný. [9] Obecný model lidského těla rozšiřuje znalosti o výrobních procesech a pomáhá zajistit správnou funkčnost již při návrhu. Velice důležitým prvkem je také zohlednění faktorů ovlivňujících ergonomii a bezpečnost práce již v počátečním vývoji a přípravě výroby. [14]

Jednou z možností je využití simulačního **SW Tecnomatix Jack** od firmy UGS. K řešení ergonomických otázek na pracovišti je použit digitální, biomechanicky přesný, model člověka s přirozenými pohyby, rozsahy kloubů a reálnými vlastnostmi, založený na studiích NASA. Model je umístěn do pracovního prostředí a následnou analýzou lze zjistit, co vidí, kam dosáhne, kdy se unaví a mnoho dalších informací. U digitálního modelu lze nastavit různou velikost, váhu, pohlaví, stáří jedince, dokonce i stupeň obezity a následně analyzovat jeho výkon. Pomocí nástroje „Low back Compression Analysis“ se automaticky vypočte zatížení v bederní části zad a následně se porovná s maximální dovolenou hodnotou z normy. Tímto lze zahrnout lidský faktor již do návrhu konstrukce, technologie výroby, montáže i servisu. Kromě konkurenceschopnosti a spokojenosti zákazníka či pracovníka má toto zásadní vliv i na bezpečnost práce, výrobní čas, výslednou kvalitu a na celkové náklady vývoje a výroby. Díky simulaci je možné vyvarovat se chyb a budoucích nákladných změn. Model pracovního prostředí lze vytvořit buď přímo v „Jackovi“, nebo se geometrie importuje z běžných CAD systémů. [15]



Obr. 3.4 - ukázka Tecnomatix Jack [O12]

Další variantou je použití SW Catia (Delmia) V5 – Human, což je produkt dalšího světového dodavatele řešení pro oblast řízení životního cyklu výrobku (PLM – Product Lifecycle Management), společnosti Delmia Systèmes Group. Typické využití tohoto SW je konstrukce automobilů. Využívá se předností systému při navrhování ergonomického interiéru vozu tak, aby řidič i cestující měli v dosahu a dohledu všechny ovládací prvky, aby měli dostatek místa a pohodlí s ohledem na antropometrii postavy. Díky tomu se daří optimalizovat ergonomické parametry a tím se více přiblížit požadavkům na snadné ovládání, bezpečnost a vysokou výkonnost. Nechybí ani analýzy pro manipulaci s břemeny (zvedání, pokládání, tlačení, tahání a přenášení), analýzy pro nalezení pohodlných poloh a statické (dynamické) analýzy zatížení jednotlivých tělesných partií. Další důležitou oblastí PLM je servis. Systém umožňuje analyzování rozebíratelnosti a výměny dílů s ohledem na lidskou obsluhu. [16]



Obr. 3.5 - ukázka Delmia V5 Human [O13]

Poslední možnost využití VH v průmyslu, kterou si zde uvedeme, je virtuální trénink a údržba. Jedním z nejlepších způsobů, jak naučit pracovníky zvládat složité a důležité úkoly, je ukázat jim film s odborníky v akci. Natočit instruktážní video s reálnými lidmi není jednoduché ani levné. Někdy to může být i nebezpečné, jestliže se jedná o zvládání krizových situací,

jako je např. nehoda v jaderné elektrárně. V takovém případě se přímo nabízí využít rozšířené reality (Augmented Reality) s VH. Nebezpečné postupy mohou být nasimulovány, zatímco reálné obrázky a videa slouží jako pozadí. [2] Pro lepší pochopení simulovaného postupu lze scénu prohlížet z různých úhlů pohledu, což je velkou výhodou oproti běžnému filmu. Precizní plánování oprav a zvládnutí havarijních stavů je dnes jednoznačně nutné, aby bylo možné dodržet vysoké bezpečnostní standardy.

### 3.4 Aplikace ve zdravotnictví

Medicína je jednou z oblastí, kde se VR stále ve větším měřítku uplatňuje jako prostředek pro vzdělávání, trénink a plánování chirurgických zákroků. Speciálně v chirurgii se s úspěchem využívá anatomicky přesných modelů člověka. Přesná data se získávají za pomoci dnes běžně používaných technologií jako je Computer Tomography (CT), Magnetic Resonance Imaging (MRI), ultrazvuk apod., která jsou pomocí počítačových technik (rendering a modelling) převedena do 3D modelu. Velkou výhodou takto získaného modelu je možnost lepšího pochopení vazeb mezi jednotlivými aspekty anatomie.



Obr. 3.6 - ukázka Techniques of Medical VR [O14]

Uživatel může model opakovaně rozebírat, skládat a prohlížet z libovolného úhlu, což v reálu samozřejmě možné není. Existují celé virtuální anatomické atlasy, pomocí kterých se vzdělávají medicí, či trénují i zkušení profesionálové se zaměřením na ověření nejnovějších operačních technik. Lékař si na takovém modelu může bez účasti pacienta detailně naplánovat či dokonce nasimulovat připravovaný zákrok, aby našel nejlepší a nejšetnější postup. Velice oblíbené jsou tréninkové simulátory hlavně v oboru endoskopické chirurgie, protože je poměrně snadné reprodukovat ve VR omezené zorné pole. Více, než v kterémkoliv jiném odvětví, jsou zde užitečné nejmodernější technické prostředky, jako jsou zobrazovací zařízení (HMD – Head Mounted Display, HUD – Head Up Display, CAVE – Computer Aided Virtual Environment, Responsive Workbench, ...) a trackovací zařízení (Datagloves, Joysticks, ...). [17]

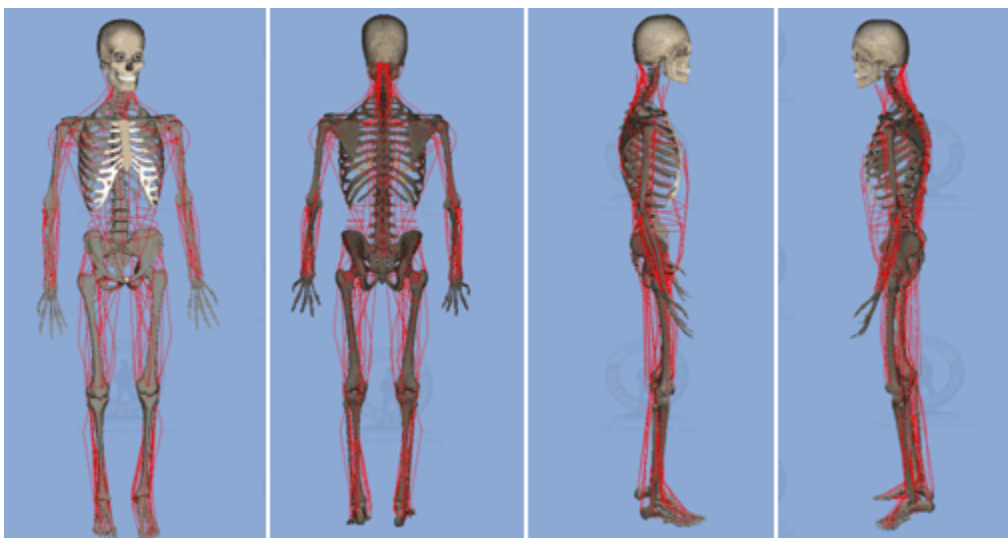
### 3.5 Aplikace v armádě

Dnešní voják bojuje ve velice drsných podmínkách, nosí těžké brnění a spoustu sofistikovaných zařízení. Existují nástroje, jak získat velmi precizní a přesnou zpětnou vazbu o navrhovaných změnách ještě ve fázi vývoje. Americká armáda spolupracuje s řadou komerčních společností a výzkumných center na modelování a simulaci virtuálních vojáků a na tvorbě aplikací schopných předpovídat lidský pohyb.



Obr. 3.7 - ukázka Virtual Soldier Research [O15]

University of Iowa se společností Santos Human Inc. pracuje v projektu Virtual Soldier Research na virtuálním prostředí Santos™, kde bude možné vytvořit novou generaci digitálního člověka, zahrnující realistické modely včetně anatomie, biomechaniky, fyziologie a inteligence v reálném čase. Jejich cílem je testování výrobků a systémů dříve, než jsou postaveny, čímž se sníží náklady a čas spojené s výrobou prototypů. Dalším aspektem je hodnocení lidského těla během provádění úkolu. SW obsahuje širokou škálu simulačních nástrojů pro testování designu, posuzování výkonu, predikci pohybu, držení těla či ergonomickou analýzu. Přístup založený na kinematické a dynamické analýze umožňuje postavě reagovat téměř nezávisle, na rozdíl od animace s uloženými daty. [18] Na obrázku 3.8 je model kostry a svalových linií vyvinutý během projektu. Tento SW má své uplatnění nejen v armádě a jeho partnery jsou světoznámé společnosti z různých odvětví (Honda, Hyundai, Ford, NASA, Boeing,



Obr. 3.8 - ukázka Musculoskeletal model [O16]



Disney).

### 3.6 Autonomous virtual characters

Jak již bylo řečeno v úvodu, právě virtuální lidé jsou jedním z důležitých aspektů k vytvoření přesvědčivějšího dojmu přítomnosti uvnitř virtuální reality. Protože, jak bychom mohli uvěřit, že jsme součástí světa, kde nejsou žádní obyvatelé? I přes rychlý technologický vývoj v oblasti hardwaru, který můžeme již dlouho sledovat, je třeba přemýšlet o virtuálních světech, které by obývaly postavy schopné spolupracovat, vyjednávat, komunikovat a jednat na základě emocí, obav a nálad. Chceme-li tedy vytvářet uvěřitelné VH, musejí být schopni zapojit se do sociálních interakcí jak s ostatními postavami, tak především s uživateli. Z tohoto důvodu musí být behaviorální model postavy dostatečně univerzální, aby umožňoval změny chování na různorodé podněty. V ideálním případě by postava měla jednat nepředvídatelně a zdánlivě samostatně.

Aby postava jednala uvěřitelně, musí být schopna vnímat své okolí a být schopna vybrat adekvátní reakci. V této souvislosti se často používá termín Autonomous Virtual Character (AVC), tedy samostatná virtuální postava. Pro realizaci vnímání musí mít AVC senzory, které simulují funkci lidských smyslů, jako je zrak a hmat. Tyto smysly by měly být základem pro reakce na zvuky, pohyby a manipulaci s objekty. Na druhou stranu je nutné, aby postava měla i vlastní paměť a byla schopna na obdobné podněty reagovat podobně. Nejzákladnější spontánní reakce člověka na externí podněty je reflex. Typickým příkladem je míč letící směrem k postavě. V takovém případě musí být engine schopný sledovat trajektorii objektu po celou dobu. V případě, že hrozí kolize objektu s postavou, musí postava „reflexivně“ zareagovat. [1] Příklad možných reakcí je patrný z obrázku 3.9.



Obr. 3.9 - ukázka reflexivního jednání [O17]

Velmi důležitým aspektem je nonverbální komunikace, pomocí které lze uvěřitelnost VH posunout na další úroveň. Je až překvapivé, jak efektivní je využití výrazů tváře, gest a různých postojů při komunikaci s virtuální postavou.

Lze tedy konstatovat, že pomocí dnešních technologií a znalostí lze vytvářet velmi zdařilá a propracovaná virtuální prostředí včetně obyvatel. Nicméně pro realizaci naprosto přesvědčivých AVC musí dojít ještě ke značnému pokroku. Jedním z předních světových výzkumů v této oblasti je Virtual Soldier Research zmíněný v kapitole 3.1.5.

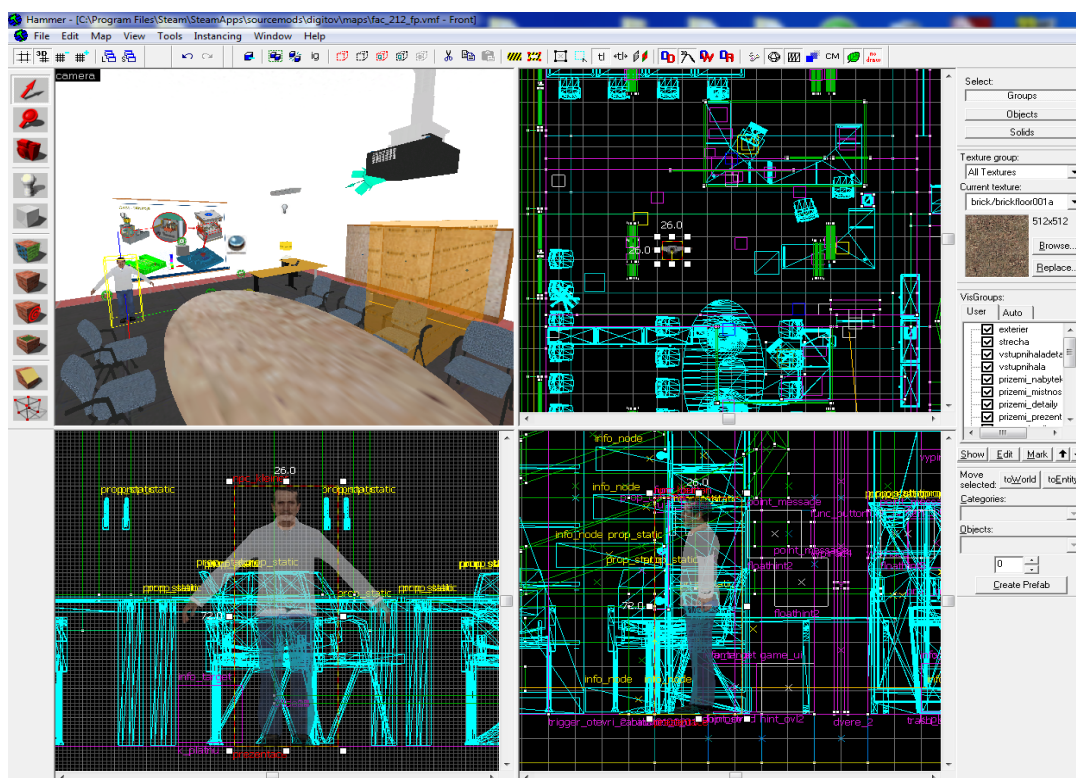
## 4 Popis vybraného nástroje

Nástrojů pro 3D animaci postav není na trhu mnoho, jako příklad lze uvést Autodesk MotionBuilder nebo Game Blender. Hlavní rozhodovací kritéria pro výběr SW byla možnost využití na KPV a pořizovací cena. Po prostudování informací o jednotlivých programech bylo rozhodnuto jednoznačně ve prospěch Face Poseru, který je součástí již využívaného SW balíčku Source SDK a náklady na pořízení jsou téměř nulové. I ostatní programy disponovaly vlastním enginem, avšak zásadní výhodou aplikace Face Poser jsou integrované knihovny modelů virtuálních postav. Odpadla tedy nutnost vytvářet vlastní postavy a i z tohoto důvodu se vybraný nástroj výborně hodí pro výukové účely. Dále následuje popis SW balíčku a speciálně součástí Face Poser.

### 4.1 Source SDK

Source SDK (Software Developer's Kit – sada nástrojů pro vývojáře) je balíček programů, který umožňuje tvorbu virtuálních prostředí včetně implementace virtuálních lidí. Tento software je volně dostupný při pořízení některého z herních titulů firmy Valve Software, přes distribuční službu Steam. Jako technologická základna pro tvorbu slouží Source Engine, na kterém běží dnes již kultovní hra Half-Life 2.

Součástí balíku Source SDK je **Valve Hammer Editor 4.1** (VHE), což je software primárně určený pro tvorbu map k počítačovým hrám a jejich následnou kompilaci. Jak se však ukázalo, může být i velmi vhodným nástrojem pro modelování výrobních prostor, či téměř čehokoliv jiného. Tento modelovací program umožňuje zobrazení ve 2D a 3D pohledech zároveň, což spolu s intuitivním ovládáním činí tento program vhodný pro výukové účely. Dále je v tomto balíku dostupný Model Viewer, což je prohlížeč jednotlivých modelů dostupných pro použití ve VHE. Pro tuto práci je však nejdůležitější součástí Source SDK nástroj **Face Poser**, který bude popsán v následující kapitole 4.3.



Obr. 4.1 - ukázka Valve Hammer Editor 4.1

## 4.2 Adresářová struktura

Součástí Source SDK je více nástrojů pro tvorbu dílčích komponent. Některé další nástroje je možné doinstalovat. V každém případě je třeba znát, kde jsou fyzicky dílčí komponenty uloženy. Každé virtuální prostředí (např. hra Half-Life 2) nebo modifikace, které je vytvořeno v Source Enginu, obsahuje vlastní adresářovou strukturu, která obsahuje např. kromě jiného tyto adresáře:

**Maps** – zde jsou uloženy všechny mapy (formát .bsp)

**Materials** – obsahuje všechny využití textury, včetně popisného textového souboru, který udává vlastnosti materiálu dané textury (formát .vtf, .vmt)

**Models** – obsahuje 3D modely (exportovaný formát .smd). Na tyto modely mohou být použity speciální textury z adresáře Materials

**Scenes** – skriptované scény (choreografie), které je možné vytvářet v programu Face Poser

**Sound** – zvuky (formát .wav)

Tato struktura může být případně kompilována v archivu .gfc. U příslušné modifikace je vždy možno uvést, z jakého balíčku (hry) jsou jednotlivé prvky „odděděny“ (inherited). Jinými slovy je možno zvolit předka dané modifikace. Z tohoto předka je možné v modifikaci využít všechny soubory.

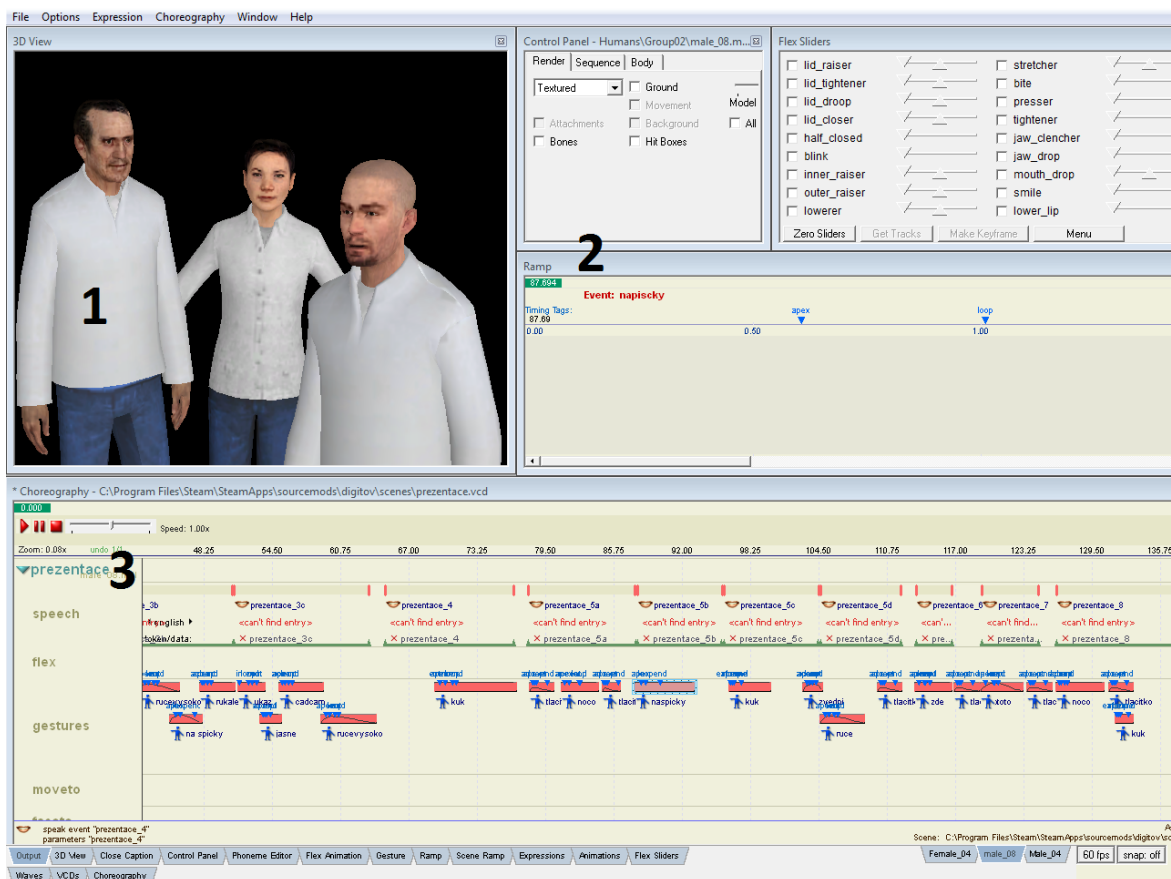
## 4.3 Face Poser

Face Poser je nástroj z balíčku Source SDK, který umožňuje tvorbu a úpravy choreografických scén pro Source Engine. Pomocí tohoto softwaru je možné zpracovávat zvukové soubory včetně synchronizace mimiky herce s přehrávanou zvukovou stopou (lip-syncing), tvořit výrazy tváře a ovládat pohyby a gesta VH pomocí skeletální animace. Hercem ve vytvářené scéně je tzv. NPC (non-player character). Každá scéna může mít herců více. Pomocí Face Poseru se vytvoří soubor s příponou .vcd, který je následně třeba implementovat do virtuálního světa pomocí VHE, aby mohla být tato scéna spuštěna.

Pod pojmem choreografická scéna si můžeme představit sadu instrukcí uloženou ve zdrojovém souboru, kde je nadefinováno chování a interakce jednotlivých NPC. Možnosti vyvolání scény v mapě budou vysvětleny dále. Stejně, jako ostatní součásti Source SDK, má i Face Poser velice uživatelsky příjemné pracovní prostředí, které si lze individuálně nastavit podle potřeby. V levé horní části okna se nachází *3D View* (1), kde se zobrazuje model postavy, která je vybrána pro aktuální scénu. V tomto okně lze s modelem manipulovat a prohlížet si tak animaci z libovolného úhlu. Velice užitečná je možnost zobrazení více modelů zároveň. Nastavení zobrazení modelu najdeme v okně *Control Panel*. V další části okna je oblast pro zobrazení dílčích nástrojů (2), které slouží k editaci jednotlivých vkládaných aktivit. Okno *Flex Sliders* slouží k nastavení výrazu obličeje. Nástroj *Ramp Tool* je velice užitečný pro definování síly jednotlivých dostupných gest v čase. Pomocí překrytí dvou animací tak můžeme získat animaci novou. Hlavní pracovní plochou je časová osa (3), kde dochází k načasování jednotlivých událostí. Aktuální scénu lze od začátku přehrávat klasicky pomocí červených ikon Play, Pause a Stop. Užitečnější je však využívat zeleného „jezdce“ nad těmito ikonami,

kterého můžeme posouvat pomocí myši. Takto lze přehrát přesně určenou část scény i pozpátku, což je velice užitečné při práci s delšími dialogy.

Dalším velice důležitým prvkem je nástroj *Animations*, který najdeme na spodní nástrojové liště. Po spuštění se otevře okno se všemi dostupnými animacemi aktivní postavy. Lze také animace roztřídit na gesta a postoje, či vyhledávat dle názvu. Po potvrzení výběru dojde k přehrávání v *3D View*.



Obr. 4.2 - ukázka Face Poser

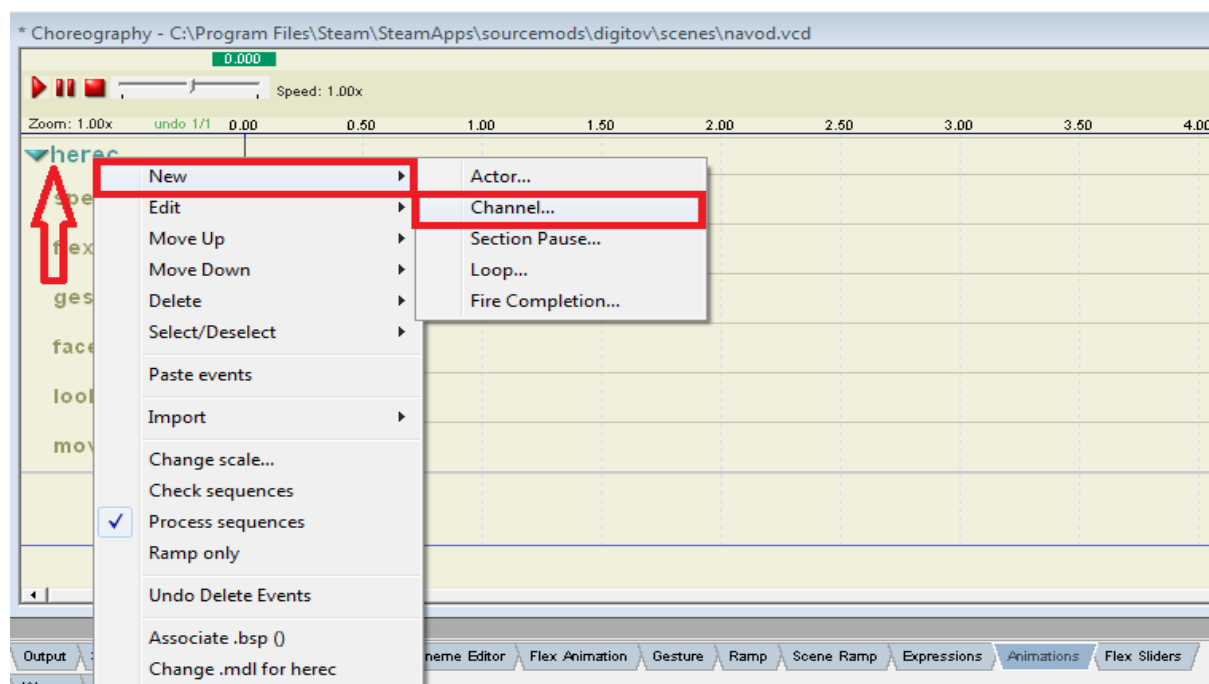


## 5 Návod pro práci ve Face Poseru

V této kapitole bude popsána práce v aplikaci Face Poser od samého začátku, to znamená od založení scény, přes nahrání zvukové stopy, až po implementaci vytvořené choreografické scény do virtuálního prostředí pomocí VHE. Hned na úvod je třeba zdůraznit, že pokaždé provedené změně ve scéně je třeba přepsat soubor *scenes.image*, aby se změna projevila. Toho docílíme jednoduše výběrem z horního menu *File>Rebuild scenes.image*. V tomto souboru jsou zkompilovány všechny soubory *.vcd* z adresáře *SCENES* (viz. Kapitola 4.2).

### 5.1 Vytvoření rámce scény

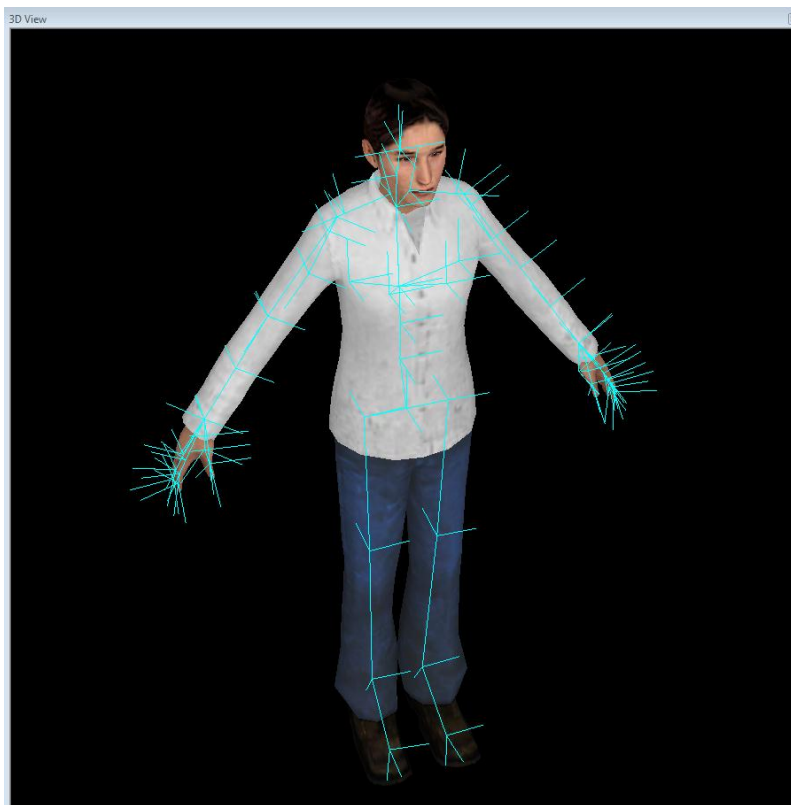
Po spuštění aplikace Face Poser z hlavního menu Source SDK založíme novou scénu výběrem z horního menu *Choreography>New* a uložíme do adresáře *SCENES*. Pro lepší orientaci a zvýšení přehlednosti je nejprve nutné vytvořit rámec scény. Každá scéna bude rozdělena na jednotlivé pojmenované postavy (herce) a každá postava bude obsahovat „kanály“ pro vkládané události. Postavě lze přiřadit libovolný počet kanálů, do kterých události vkládáme. Je vhodné každému herci přiřadit kanál na řeč (speech), animaci obličeje (flex), gesta (gesture), pohyb (moveto), orientaci postavy v prostoru (faceto) a pohled (lookat). Kanály se přidávají stisknutím pravého tlačítka myši (**RC – right click**) na jméno herce v časové ose a výběrem *New>Channel* (obrázek 5.1).



Obr. 5.1 - ukázka vytvoření nového kanálu (Face Poser)

### 5.2 Načtení modelu postavy

Po založení nové scény je třeba vybrat model postavy, přes hlavní menu *File>LoadModel* v adresáři *MODELS\HUMANS\*. Po stisku *Open* se model zobrazí v okně 3D view, kde je možno s ním manipulovat dle tabulky 5.1. V okně Control Panel lze nastavit zobrazení kostry modelu a texturu, či přehrávat dostupné animace. Aby postava přehrávala dále vkládané události, musí být model člověka asociován s hercem pomocí RC kdekoli na časové ose výběrem *Associate .mdl with <jméno herce>*.



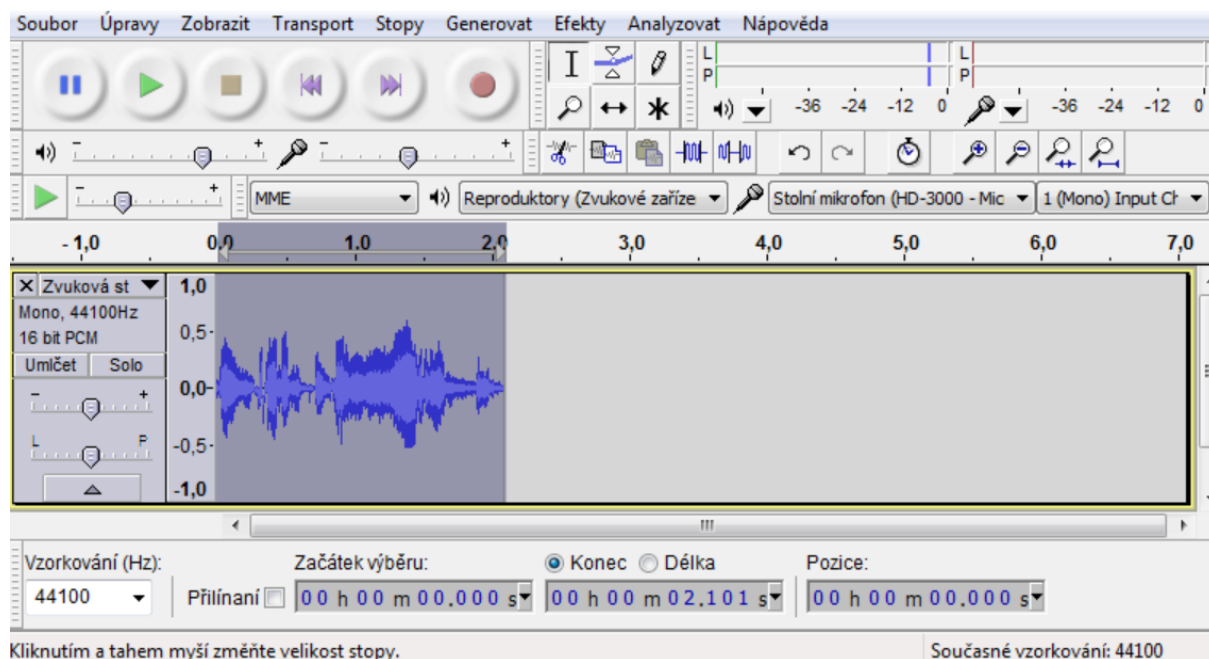
Obr. 5.2 - zobrazení modelu v 3D View (Face Poser)

MOUSE2 + tažení myši	Zoom
MOUSE1 + tažení myši	Otáčení v prostoru
SHIFT + MOUSE1 + tažení myši	Posun v prostoru

Tabulka 5.1 - manipulace s modelem v 3D View

### 5.3 Příprava a vložení zvukové stopy

Základním prvkem choreografické scény je zvuk, tedy přesněji řeč. Pomocí virtuální postavy, která k nám promlouvá, dojde ke značnému oživení virtuálního prostředí. Zvuková stopa, kterou chceme využít v aplikaci Face Poser, by měla být ve formátu *4-bit Microsoft ADPCM mono s příponou .wav a s frekvencí 44100Hz*. Případně lze využít i *16-bit nekomprimovaný formát PCM*, ale je třeba počítat s větší velikostí souboru. Pro záznam zvuku lze využít jakýkoliv zvukový editor a rekordér, například volně dostupný program Audacity. Po nahrání a případném vyčištění zvuku stačí výběr exportovat do výše popsaného souboru a uložit do složky *SOUND* v adresáři editovaného módu.



Obrázek 5.1 - ukázka Audacity 2.0.2

Když máme připravené všechny dialogy, musíme ještě vytvořit tzv. soundscript. Tento záznam, ve formě textového souboru s příponou *.txt*, identifikuje, kde lze příslušný zvukový soubor najít a jak s ním naložit. Jméno skriptového souboru je libovolné a dále zbývá skript implementovat do souboru *game\_sounds\_manifest.txt* v adresáři *SCRIPTS*, který je automaticky načítán při spuštění Face Poseru. Příklady skriptů jsou uvedeny v příloze. Po uložení skriptů jsou již všechny zvuky přístupné a připravené k použití.

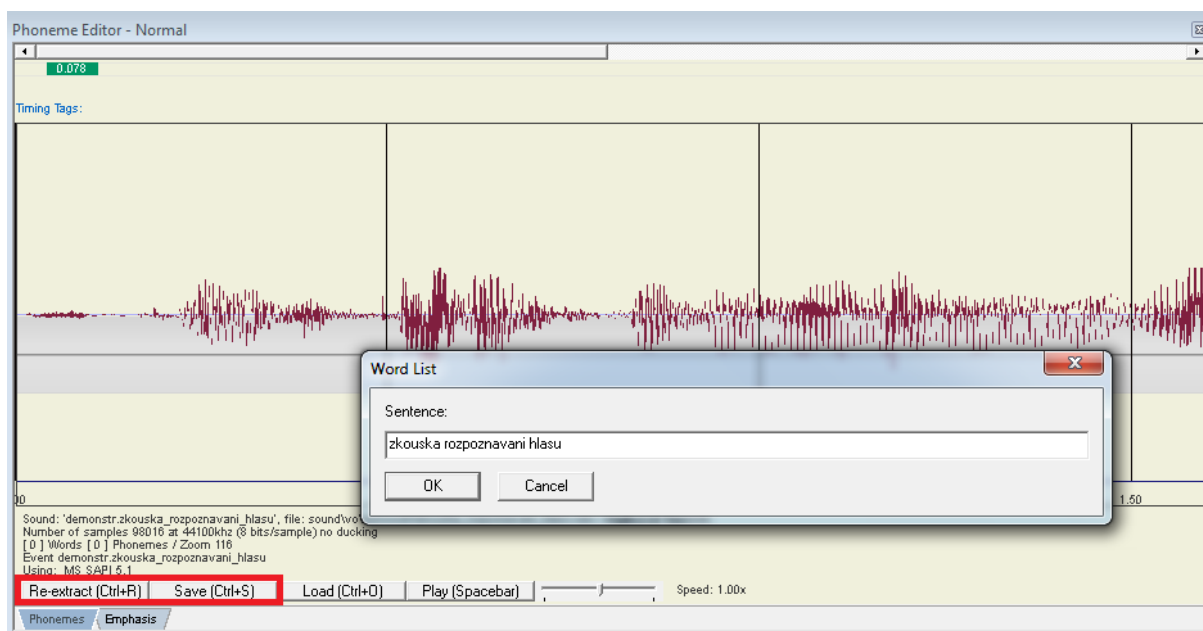
Vložení zvukové stopy se provede stisknutím pravého tlačítka myši v oblasti řeči (speech) na časové ose, výběrem *WAV File*. Zobrazí se okno, ve kterém vybereme název příslušného souboru, případně změním hlasitost, která je již ale nastavena v soundscriptu. Po vložení můžeme stopu libovolně posouvat po časové ose pomocí uchopení myší.

### 5.3.1 Phoneme Tool

Po vložení audio souboru je nutné provést synchronizaci mimiky modelu s přehrávanou zvukovou stopou. K tomu slouží nástroj Phoneme Editor, který najdeme na spodní nástrojové liště (viz obr. 4.2). Tento nástroj vyžaduje nainstalovaný software Microsoft Speech API 5.1, společnosti Microsoft. Program provede automatické rozpoznání řeči a uloží do zvukového souboru informace o umístění jednotlivých hlásek (fonémů). Software je schopný bezchybně rozpoznat bohužel pouze anglofonní jazyky, takže při rozpoznávání českého jazyka je většinou nepřesný. Navíc na oficiálním webu firmy Valve Software se dozvíme, že automatické rozpoznání řeči je nefunkční na Windows Vista a vyšších. Pro vlastní tvorbu firma využívá odlišné knihovny pro rozpoznávání, které bohužel neplánuje distribuovat jako součást Source SDK. [19] Na internetu lze nalézt různé opravné balíčky, avšak nelze zaručit funkčnost na všech systémech. Jako nejspolehlivější řešení se jeví používat nástroj Phoneme Tool na operačním systému Windows XP.

Po označení zvukové stopy v kanálu řeči otevřeme Phoneme Editor a stiskneme tlačítko *Re-extract* (Ctrl + R), otevře se dialogové okno, do kterého přepíšeme doslovně text nahrávky (bez diakritiky a interpunkce) a stiskneme *OK*. Ve většině případu se objeví hláška "*Last Extraction Result: results retrieved, but full recognition failed*" - přeloženo obdrženo výsledek, ale

plné rozpoznání selhalo. V praxi to znamená, že synchronizace není úplně přesná a některá slova se nerozpoznají.



Obrázek 5.2- ukázka Phoneme Editor (Face Poser)

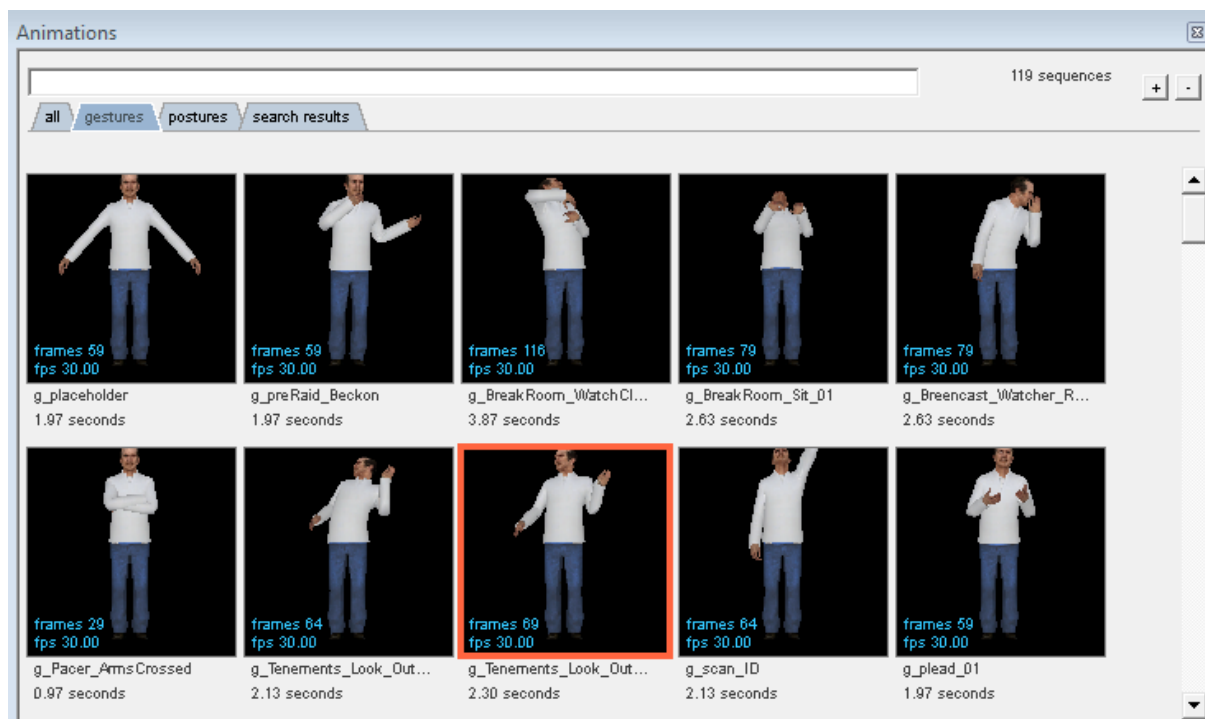
Aby se změny projevily na modelu, je třeba pomocí RC v oblasti zvukové stopy vyvolat nabídku, stisknout *Commit Extraction* a poté uložit pomocí *SAVE* (Ctrl + S). Když extrakce fonémů neproběhne bezchybně, lze jednotlivé hlásky posouvat, odstraňovat či přidávat manuálně, viz tabulka 5.2. Při manipulaci s jednotlivými hláskami se může objevit kritická chyba a program se sám ukončí. Nelze než doufat, že programátoři z Valve Software zapracují na těchto slabínách, které mohou práci značně znepríjemnit. Na druhou stranu je třeba uvést, že ostatní funkce Face Poseru fungují bezchybně. Jako náhradní řešení lze použít vkládání výrazů obličejů do kanálu Flex, což plnohodnotně nahradí chybějící hlásky. Při práci s dlouhými dialogy se musí počítat s vyšší časovou náročností. Vložení vybraného výrazu obličejů se provede, po otevření nástroje Expressions na spodní liště, přetažením animace pomocí myši na časovou osu.

TAB/SHIFT + TAB	Označení slov vpřed/vzad
SHIFT + šipky	Pohyb vybraným slovem/fonémem
INSERT/SHIFT + INSERT	Vložení nového fonému vpřed/vzad
DELETE	Odstranění výběru

Tabulka 5.2 - ovládání Phoneme Tool

## 5.4 Vkládání a úprava gest

Po vložení připravených zvuků je třeba scénu doplnit vhodnými gesty. Na spodní nástrojové liště vybereme položku Animations. V otevřeném okně (obrázek 5.3) vybereme vhodnou animaci, kterou následně přesuneme pomocí myši do kanálu *gesture*. Animace lze třídit na postoje (postures), gesta (gestures) či vyhledávat podle názvu. Při stisknutí MOUSE 1 na ikonu animace dojde k přehrání v *3D View*.



Obrázek 5.3 - ukázka dostupných gest pro model male\_08.mdl (Face Poser)

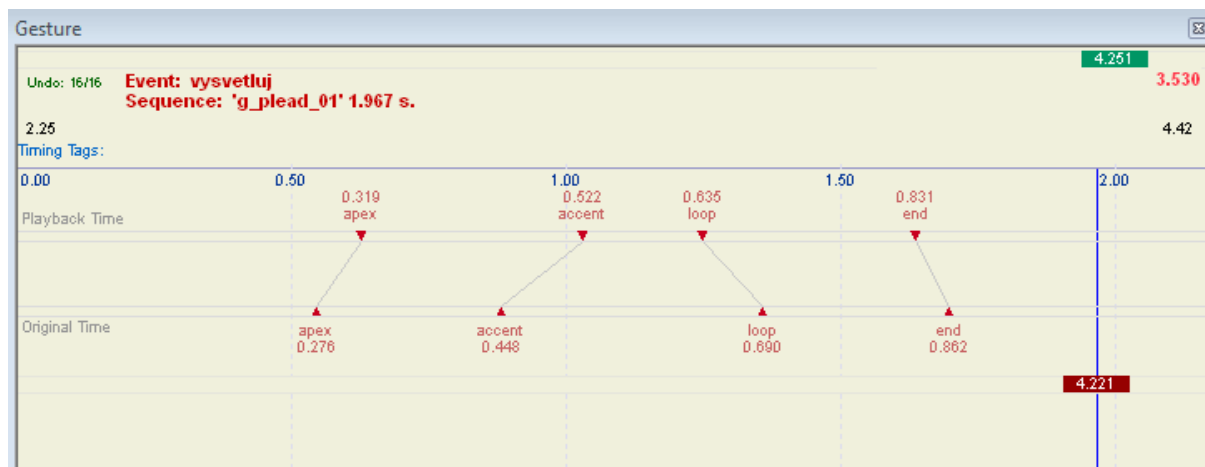
Každé gesto se skládá ze 4 základních částí, tzv. „Timing Tags“, které můžeme posouvat po časové ose v nástroji Gesture (obrázek 5.4) a tím modifikovat vybranou animaci. Pomocí změny časování má designér scény možnost ovlivnit, jak bude animace v Source Engineu vypadat.

*Apex* – začátek animace

*Accent* – konec hlavní animace a začátek zdůraznění

*Loop* – konec zdůraznění

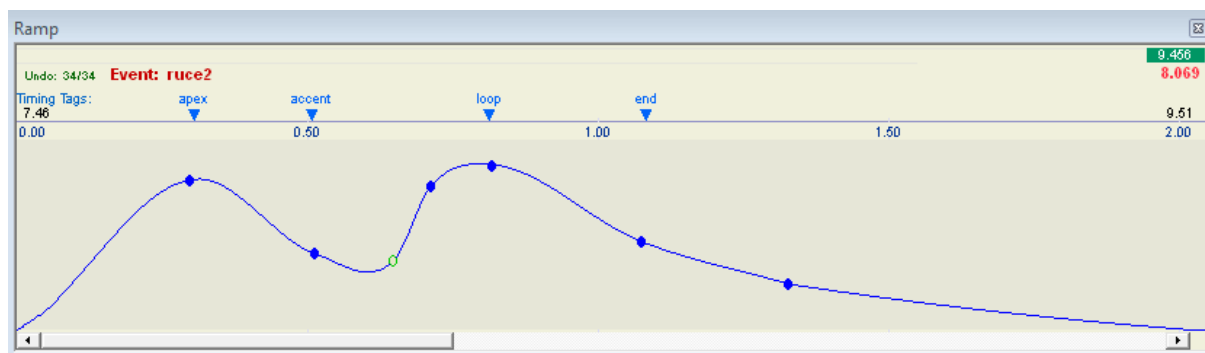
*End* – postupné ukončení celé animace



Obrázek 5.4 - ukázka Gesture Tool (Face Poser)

### 5.4.1 Ramp Tool

Další užitečnou pomůckou je nástroj Ramp Tool (obrázek 5.5), který se automaticky načte při spuštění Face Poseru. Pomocí této aplikace se určuje absolutní hodnota každé události v čase, což se výborně hodí pro „doladění“ vkládaných gest. Celá manipulace probíhá pomocí grafu dvou proměnných – času a intenzity. Pro vytvoření bodu na grafu stačí stisknout CTRL+LC. Takových bodů můžeme vytvořit libovolné množství a následně je možné s nimi manipulovat pomocí levého tlačítka myši. Díky tomu, že software automaticky vygeneruje křivku v okolí bodu, docílíme přirozeného pohybu.

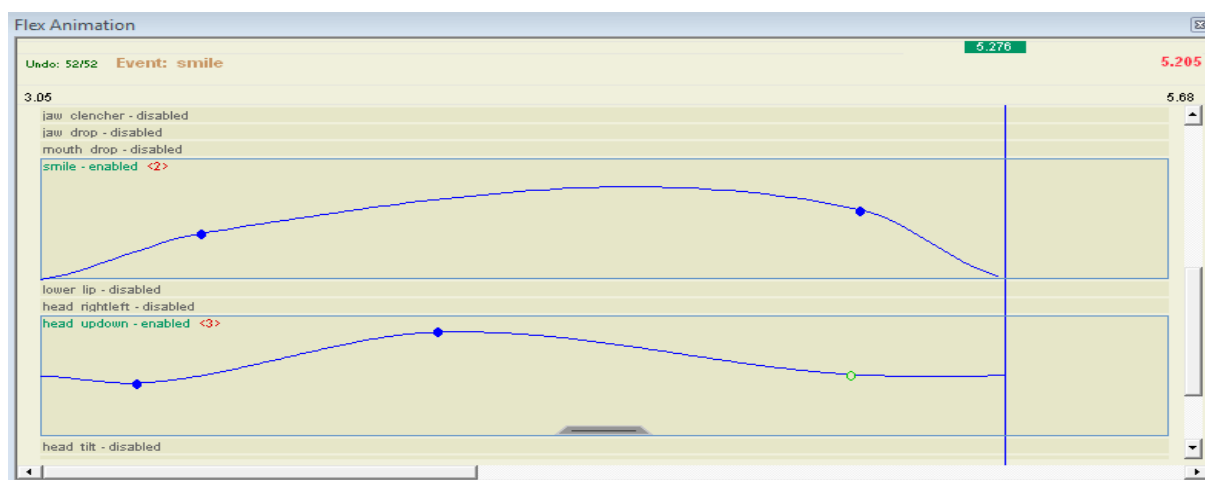


Obrázek 5.5 - ukázka Ramp Tool (Face Poser)

## 5.5 Animace obličeje

K modifikaci výrazu obličeje slouží ve Face Poseru nástroj *Flex Animation*, který najdeme opět ve spodní nástrojové liště. Tento nástroj využívá metodu per-vertex animace, což je metoda přímé manipulace s vrcholy modelu, bez použití skeletu. Tento postup se výborně hodí právě pro animaci obličeje, ale lze ho využít v kterékoliv části modelu.

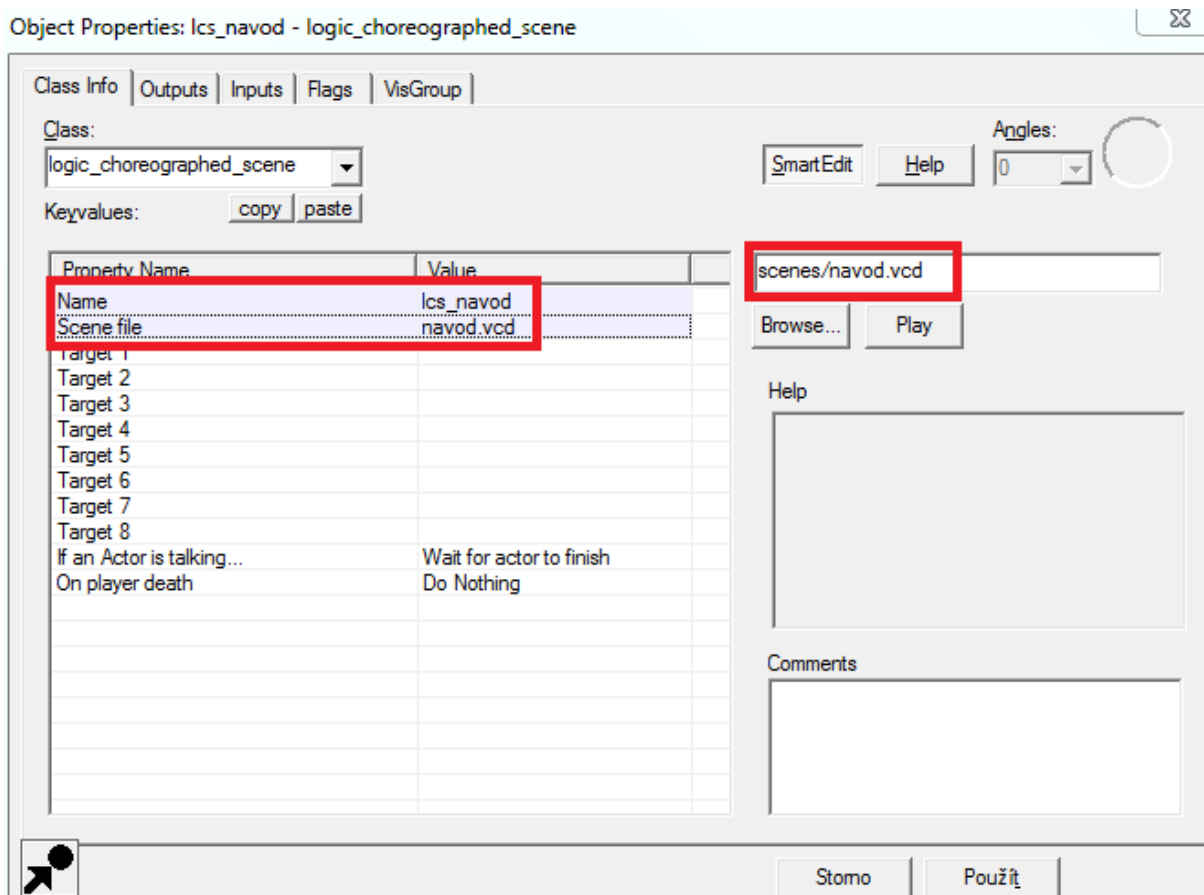
Událost, pro úpravu výrazu, vložíme pomocí RC v oblasti kanálu *flex* výběrem *Flex animation* z nabídky. Událost nejprve pojmenujeme a stiskneme *OK*, následně označíme pomocí myši a otevřeme nástroj *Flex Animation*. Ze seznamu si vybereme část obličeje, kterou chceme modifikovat a pomocí „double-click“ vyvoláme graf, se kterým pracujeme analogicky jako s nástrojem Ramp Tool. Na obrázku 5.6 je vidět vytvoření úsměvu a pokývání hlavou.



Obrázek 5.6 - ukázka Flex Animation (Face Poser)

## 5.6 Implementace scény pomocí VHE

Pro další práci je třeba scénu implementovat do mapy pomocí VHE. Po otevření Hammer Editoru načteme mapu, kterou máme v plánu modifikovat. Nejprve vložíme NPC pomocí *Model Browseru* nebo pomocí *Entity Tool*. Nejvhodnější je použít class *NPC\_ALYX*. Důležité je, že položky *Name* a *World Model*, musí být shodné s výběrem ve Face Poseru, aby byla postava schopná přehrát scénu. Dále je třeba libovolně do prostoru vložit entitu **logic\_choreographed\_scene (LCS)**, která vytvořenou scénu přehrává. Po otevření vlastností vyplníme pole *Name*, pod kterým bude entita volána a v poli *Scene file* vybereme pomocí tlačítka *Browse* vybranou scénu (obrázek 5.7).



Obrázek 5.7 - nastavení entity LCS (VHE)

Nyní je scéna připravena k použití, zbývá pouze vytvořit entitu, pomocí které se vše spustí. Nejvhodnější je vytvořit *brush*, kterému dáme texturu *toolstrigger*, převedeme na entitu a vybereme třídu *trigger\_multiple*. V záložce *Class Info* vyplníme pouze jméno a přepneme do záložky *Outputs*. Pro vložení nového výstupu stiskneme tlačítko *Add...* a vyplníme postupně následující položky (viz obr 5.8):

*My output named* – zde máme na výběr z více možností. Entita *trigger\_multiple* testuje přítomnost hráče uvnitř svého objemu. Vybereme například *OnStartTouch*. To znamená, že jakmile hráč vstoupí do objemu entity, spustí se dále definovaný příkaz.

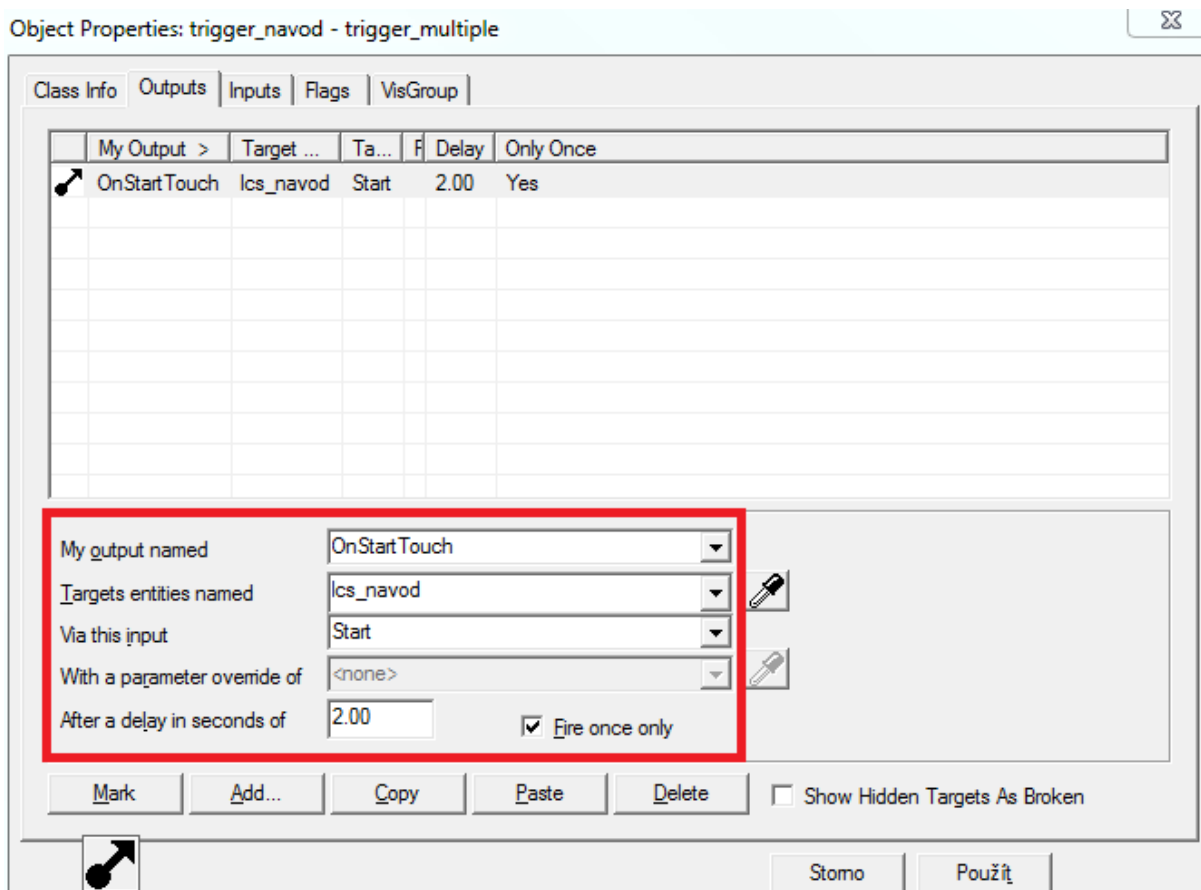
*Targets entities named* – jméno entity, kterou chceme spustit. V našem případě vyplníme jméno LCS, kterou jsme dříve vytvořili.

*Via this input* – zde vybereme ze seznamu možnost *Start*.



*After a delay in seconds of* – zde je možno vyplnit prodlevu se kterou se LCS vyvolá.

V případě, že chceme, aby se scéna mohla přehrát pouze jednou, „zaškrtneme“ možnost *Fire once only*. To je užitečné především, když se postava během scény přesune na jiné místo a opětovné přehrání by nebylo vhodné.



Obrázek 5.8 - nastavení nového výstupu (VHE)

### 5.6.1 Odezva na podnět

Další možností, jak scénu implementovat do prostředí, je využít výstup *OnPlayerUse*. Tento výstup ovšem není dostupný pro všechny třídy NPC, proto je vhodné používat výše zmíněné *NPC\_Alyx*. Pomocí tohoto výstupu se nastaví reakce postavy na stisknutí klávesy „E“, což je velice užitečné pro opakované volání scény. Vybereme postavu, která bude spouštět scénu a vytvoříme nový výstup analogicky, jako v případě spuštění přes trigger.

*My output named* – *OnPlayerUse*

*Targets entities named* – jméno entity, kterou chceme spustit. V našem případě vyplníme jméno LCS, kterou jsme dříve vytvořili.

*Via this input* – zde vybereme ze seznamu možnost *Start*.

*After a delay in seconds of* – zde je možno vyplnit prodlevu se kterou se LCS vyvolá.



## 5.7 Orientace a pohyb v prostoru

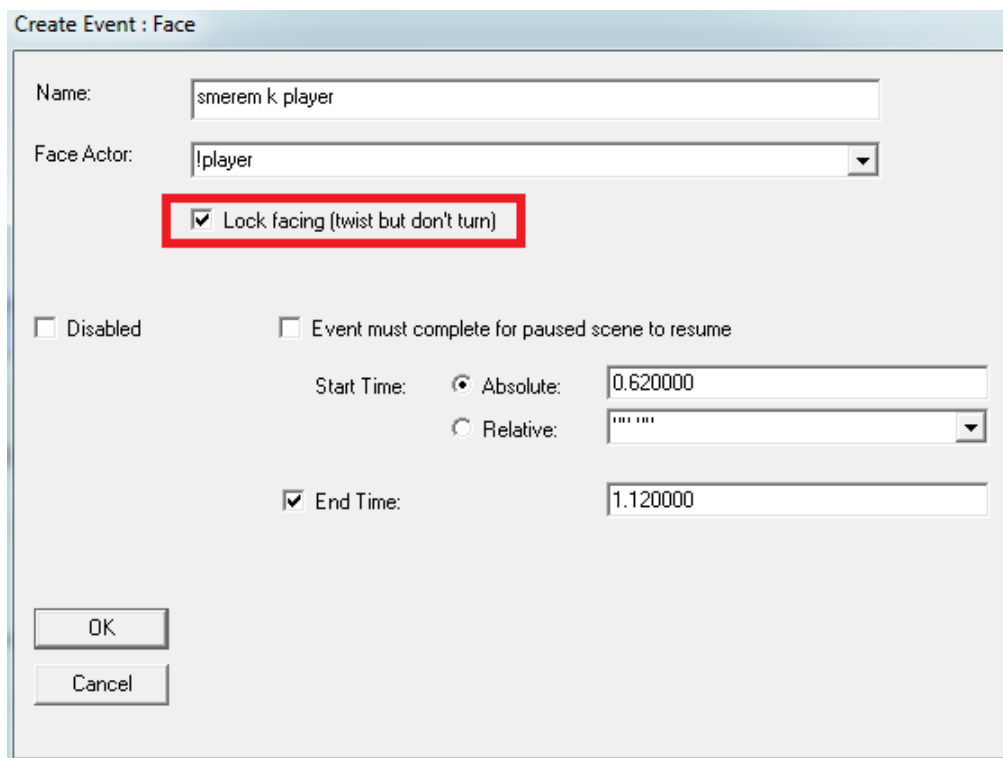
Pro nastavení pohybu a orientace v prostoru je třeba pracovat zároveň s Face Poserem a VHE, protože vždy musí být nastaveno tzv. „targetname“ (jméno cíle), ke kterému se postava otáčí či přesunuje. Tímto cílem může být jakákoliv entita našeho virtuálního prostředí, včetně hráče samotného.

### 5.7.1 Look at actor

Pomocí této události se nastaví, kam má vybraný herec otočit hlavu (nikoliv tělo). Je-li nastavený cíl v příliš velkém úhlu od postavení herce, může pohled vypadat nepřirozeně. V tomto případě je třeba událost zkombinovat s událostí *Face to actor* (kapitola 5.7.2). Pomocí RC v kanálu *lookat* vyvoláme menu a vybereme možnost *Look at actor*. V zobrazeném dialogovém okně nastavíme libovolné jméno (name) události, slouží pro snazší orientaci v událostech, a cíl, na který otočí postava svoje oči. Když potřebujeme, aby se postava podívala na místo, kde zrovna není žádná entita, můžeme jednoduše pomocí *Entity Tool* ve VHE vložit do prostoru entitu *info\_target*. Této entitě stačí vyplnit pouze jméno, aby bylo možné se na ni odkazovat.

### 5.7.2 Face to actor

Pomocí této události se nastaví, kam má herec otočit své tělo. Pomocí RC v kanálu *faceto* vyvoláme menu a vybereme možnost *Face to actor*. Další nastavení jsou naprosto analogická, jako v předchozí kapitole. Jediným rozdílem je možnost zvolit *Lock facing*, což zamezí otáčení dolních končetin. Tuto možnost využijeme v případě, kdy otočení celého těla působí nepřirozeně, nebo když chceme docílit ohlédnutí do velkého úhlu.



Obrázek 5.9 - nastavení události Face to actor (Face Poser)

### 5.7.3 Move to actor

Velice důležitou událostí je pohyb v prostoru, pomocí níž se herec může přesouvat ve virtuálním prostředí. Událost vytvoříme opět pomocí RC a výběrem *Move to actor* z nabídky. Vyplníme následující parametry:

*Name* – zvolené jméno události

*Move To Actor* – název cílové entity (např. *info\_target*)

*Walk, Run, CrouchWalk* – druh pohybu

*Stop Distance* – vzdálenost od cíle, kdy se má herec zastavit (v případě entit NPC či Player minimálně 33 jednotek)

*Force Short Movements* – herec bude postupovat po malých krůčcích

Create Event : MoveTo

Name:

Move To Actor:

Stop Distance: 40   Force Short Movements

Closest To:

Disabled  Event must complete for paused scene to resume

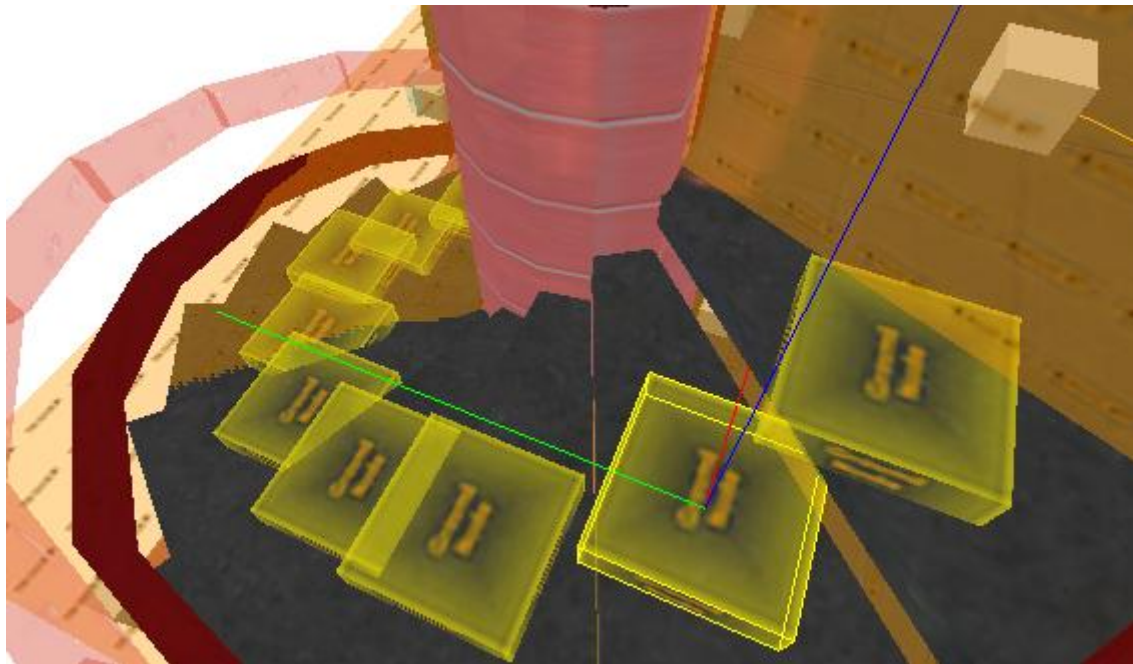
Start Time:  Absolute:   
 Relative:

End Time:

Obrázek 5.10 - nastavení události Move To Actor (Face Poser)

Pokud jsme v kolonce *Move To Actor* nevyplnili název existujícího NPC (případně !player), je třeba pomocí VHE specifikovat kam se má herec pohybovat. Nejjednodušší postup je využít entitu *info\_target*, kterou umístíme do požadované pozice. Tento postup se hodí pro jednoduchý přesun, nejčastěji v rámci jedné místnosti. Když potřebujeme, aby se herec vydal na delší trasu, je výhodnější použít entity třídy *path\_track*. Pomocí libovolného počtu „uzlů“ je možné herece navigovat téměř kamkoliv. Problém může nastat při chůzi do schodů, ale vše lze vyřešit pomocí vložení entity *info\_node* (obrázek 5.11) na každý schod. Na této entitě není co nastavovat, slouží pouze pro výpočet nejkratší trasy. Dále pokud chceme, aby herec prošel

zavřenými dveřmi, je třeba do dostatečné vzdálenosti umístit „trigger“ a přidat výstup, který je otevře.



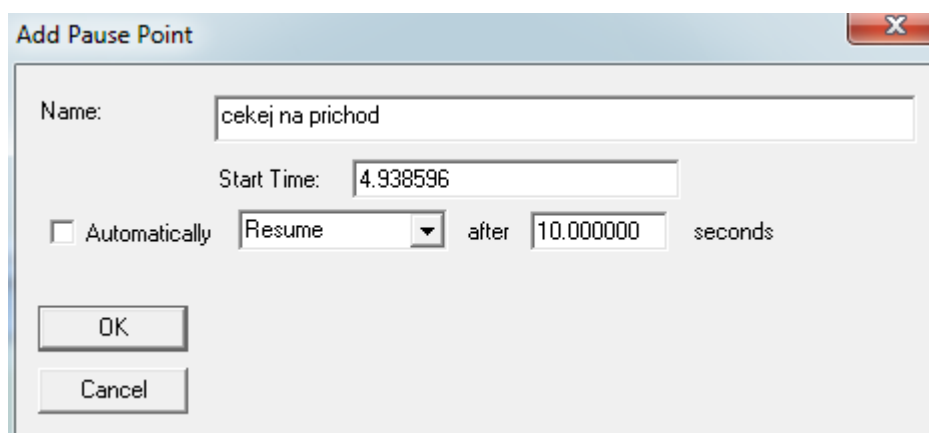
Obrázek 5.11 - vložení entit info\_node (VHE)

## 5.8 Interakce

Možnost interakce a prediktivního chování rozšiřují ve Face Poseru události *Section Pause*, *Loop*, *Sub-scene* a *Fire Trigger*. Jsou to velice jednoduché a rychle nastavitelné nástroje, které však umožňují připravit scénu na náhodné chování „hráče“ či nepředvídané situace. Obzvláště kombinací následně uvedeného se otevírají značné možnosti.

### 5.8.1 Section Pause

Událost *Section Pause* zajistí, že všechny události před bodem pauzy, které mají v nastavení „zatrženou“ vlastnost *Event must complete before paused scene to resume*, budou dokončeny, dříve než začnou události následující. Tento postup je vhodný využít v situaci, kdy se herec přesunuje v hojně „obydleném“ prostředí. Pauza vložená například před dialog zajistí, že herec začne mluvit až po příchodu do cíle. Vložení události se provede pomocí RC na časové ose výběrem *New>Section Pause*. Pauza se graficky projeví jako svislá modrá přerušovaná čára s ukazatelem, který je možno posouvat v čase pomocí uchopení myši. V dialogovém okně stačí nastavit název pauzy, popřípadě vybrat možnost *Resume* (pokračovat ve scéně), nebo *Cancel* (zrušit scénu) po uplynutí zadaného časového úseku.



Obrázek 5.12 - vložení Section Pause (Face Poser)

### 5.8.2 Loop

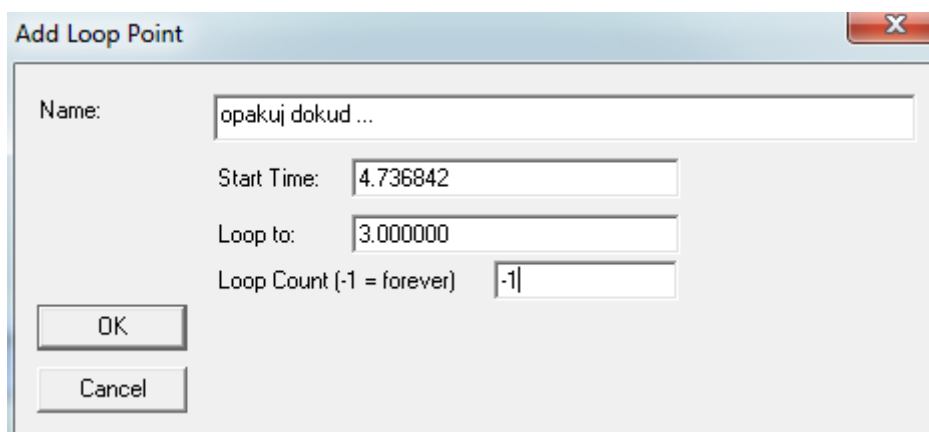
Pomocí události Loop lze vytvořit v rámci scény smyčku, která se bude opakovat. Jakmile se dostaneme do nastaveného bodu, scéna se přesune na předem určený čas. Událost se vloží opět jednoduše pomocí RC, v oblasti časové osy výběrem *New>Loop* a lze ji posouvat pomocí myši. V nastavení vyplníme následující parametry:

*Name* – libovolné jméno

*Start Time* – čas, ve kterém se smyčka spustí (lze měnit posouváním ukazatele po časové ose)

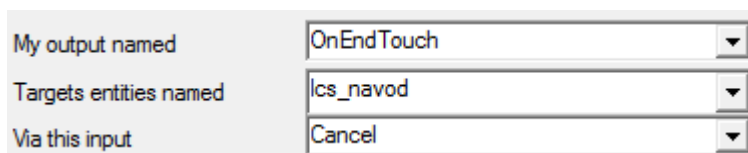
*Loop to* – čas, do kterého se scéna vrací (implicitně nastaveno na začátek scény)

*Loop Count* – počet opakování smyčky (hodnota „-1“ znamená opakovat napořád)



Obrázek 5.13 - vložení Loop Point (Face Poser)

V případě, že v kolonce Loop Count vyplníme „-1“, tedy opakovat napořád, je možné pomocí výstupu ve VHE opakování zrušit. Stačí do libovolného „triggeru“ přidat výstup a do kolonky *Via this input* zadat příkaz *Cancel*, čímž se nám dostává do ruky velice silný nástroj. Například lze nastavit, aby NPC opakoval nějakou činnost, dokud nebude splněna určená podmínka (stisknutí tlačítka, průchod dveřmi, ...).



Obrázek 5.14 - nastavení výstupu pro zrušení opakování (VHE)

### 5.8.3 Sub-scene

Další užitečnou pomůckou je vnoření dalších scén do hlavní scény. Vložení se provede pomocí RC kdekoliv na časové ose výběrem *Sub-scene* (při vkládání více scén je vhodné vytvořit samostatný kanál). V nastavení se vyplní pouze jméno a cesta k souboru scény. Pro zamezení vzniku kolizí je namísto používat tuto událost spolu se *Section Pause* (kapitola 5.8.1) a dbát na to, aby scéna vnořená neměla již žádné další vnořené scény.

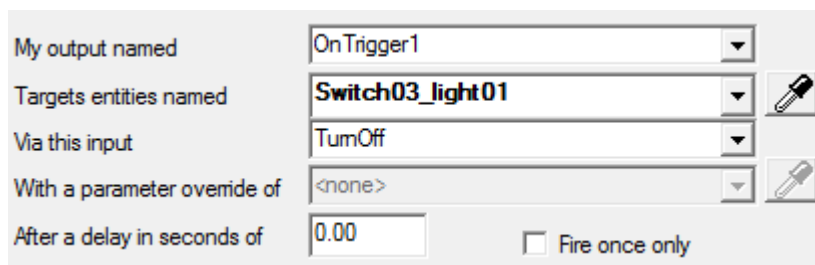
### 5.8.4 Fire Trigger

Pomocí události *Fire Trigger* je možné spouštět jakékoliv výstupy třídy *OnTrigger* ve virtuálním prostředí. Jako příklad lze uvést zhasnutí světla v přesně definovaném čase po splnění libovolných ostatních podmínek. Vložení provedeme ve Face Poseru opět pomocí RC v oblasti časové osy výběrem *Fire Trigger* z nabídky a v nastavení vyplníme jméno a číslo výstupu. Následně se přepneme do VHE a v entitě *logic\_choreographed\_scene* přidáme výstup s následujícími parametry:

*My output named* – OnTriggerX (X znamená číslo výstupu nastavené ve Face Poseru)

*Target entities named* – jméno entity, kterou chceme vyvolat

*Via this input* – příkaz pro volanou entitu



Obrázek 5.15 - zhasnutí světla pomocí Fire Trigger (VHE)

## 6 Komplexní ukázka

Cílem praktické části této práce byla implementace mluvících a reagujících postav do virtuálního prostředí, čímž mělo dojít k jeho oživení a prezentaci. Autor této práce převzal hotový model virtuální továrny od Ing. Petra Hořejšího Ph.D. a Bc. Jiřího Polcara, včetně hotové montážní linky RC modelu i s personálem a sedících postav v kancelářích.

### 6.1 Popis ukázky

Celá ukázka zapadá do žánru virtuální prohlídky na médiu virtuální realita a je určena pro posluchače předmětu KPV/DPVR. Po spuštění programu se objeví virtuální prostředí z pohledu první osoby. Ovládání je realizováno pomocí myši a klávesnice (viz tabulka 6.1). Stručná nápověda k ovládání je zobrazena ihned po načtení mapy. Důležité informace, jako jsou popisky k jednotlivým dílčím objektům (kanceláře, osoby, produkty, tlačítka atp.), se zobrazí při držení pravého tlačítka myši. To se hodí zejména při prohlídce jednotlivých dílčích částí výroby u výrobního pásu, ale také pro zviditelnění skrytých funkcí (přeskočení prezentace atp.)

Pohyb vpřed	W
Pohyb vzad	S
Pohyb vlevo	A
Pohyb vpravo	D
Běh	SHIFT
Skok	Mezerník
Přikrčení	CTRL
Použit (otevřít dveře, zvednout předmět,...)	E
Zobrazení popisu	MOUSE 2

Tabulka 6.1 - ovládání ukázky

Prvním krokem bylo vytvoření potřebných zvukových stop, včetně poměrně dlouhé prezentace systému digitální fabriky a provedení lip-syncingu. Následně byly vytvářeny choreografické scény pro jednotlivé místnosti, které se automaticky spouští většinou průchodem dveřmi pomocí entit *trigger\_multiple*. Pro větší oživení byly do prostor vloženy další postavy, které chodí po předem určené trajektorii (entity *path\_track*) v zorném poli „hráče“. Hlavní herci ve vývojovém oddělení zopakují představení pracoviště při stisknutí klávesy „E“, což je řešeno pomocí výstupu *OnPlayerUse*.

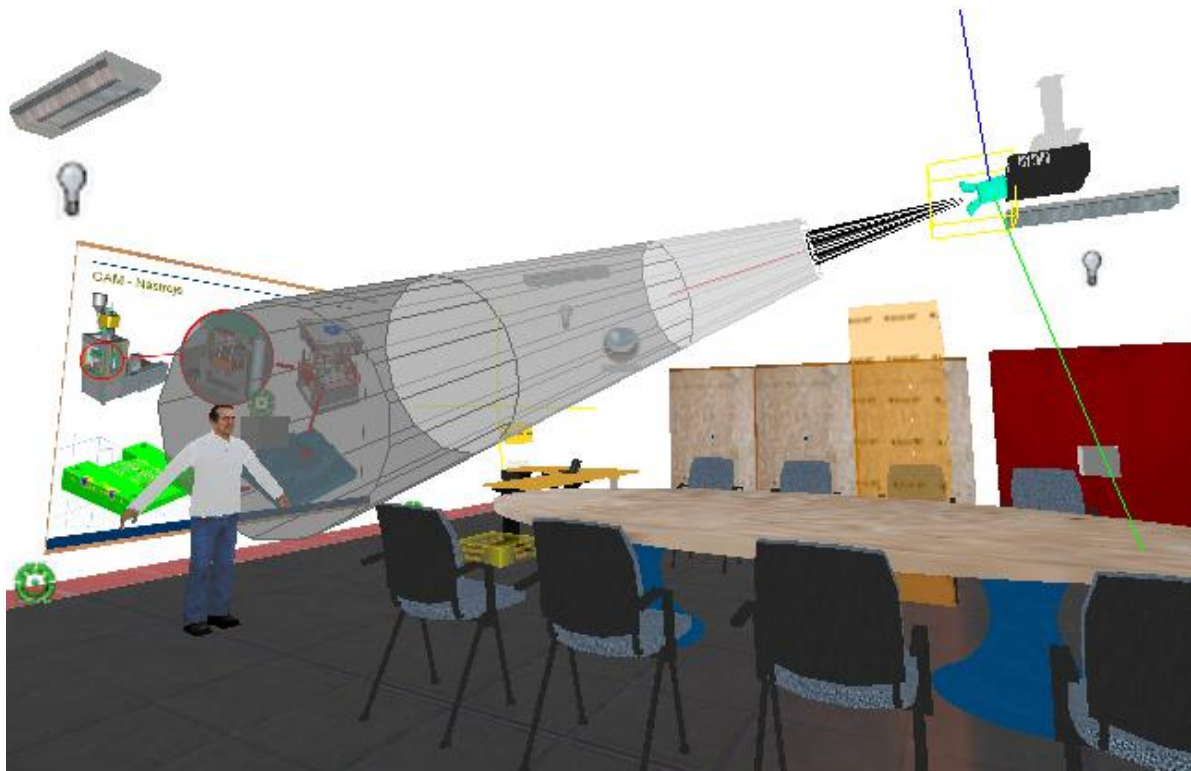
#### 6.1.1 Úvodní scéna

Úvodní scéna *lcs\_uvod* (uvod.vcd) je spuštěna průchodem hlavním vstupem pomocí entity třídy *trigger\_multiple* (*trigger\_start*). Postava „uvadecka“ (*female\_02.mdl*) uživatele přivítá, dovede ke dveřím do prezentační místnosti a vyzve ke vstoupení. Následně se herečka odebere po schodišti do prvního patra, kde čeká na spuštění další scény. Přesun po schodišti je řešen pomocí entit *path\_track* (*path\_schody\_1* až 3). Druhou postavou v úvodní scéně je „vitacka“ (*female\_04.mdl*), která se ihned po spuštění otočí čelem k uživateli a s úsměvem zamává. Tato postava reaguje na stisknutí klávesy „E“ pomocí výstupu *OnPlayerUse* zopakováním svého výstupu.

#### 6.1.2 Prezentační místnost

Průchodem dveřmi do prezentační místnosti se se zpožděním 5 vteřin spustí scéna *lcs\_prezentace* (*prezentace.vcd*), aktivuje světlo od projektoru a první snímek prezentace.

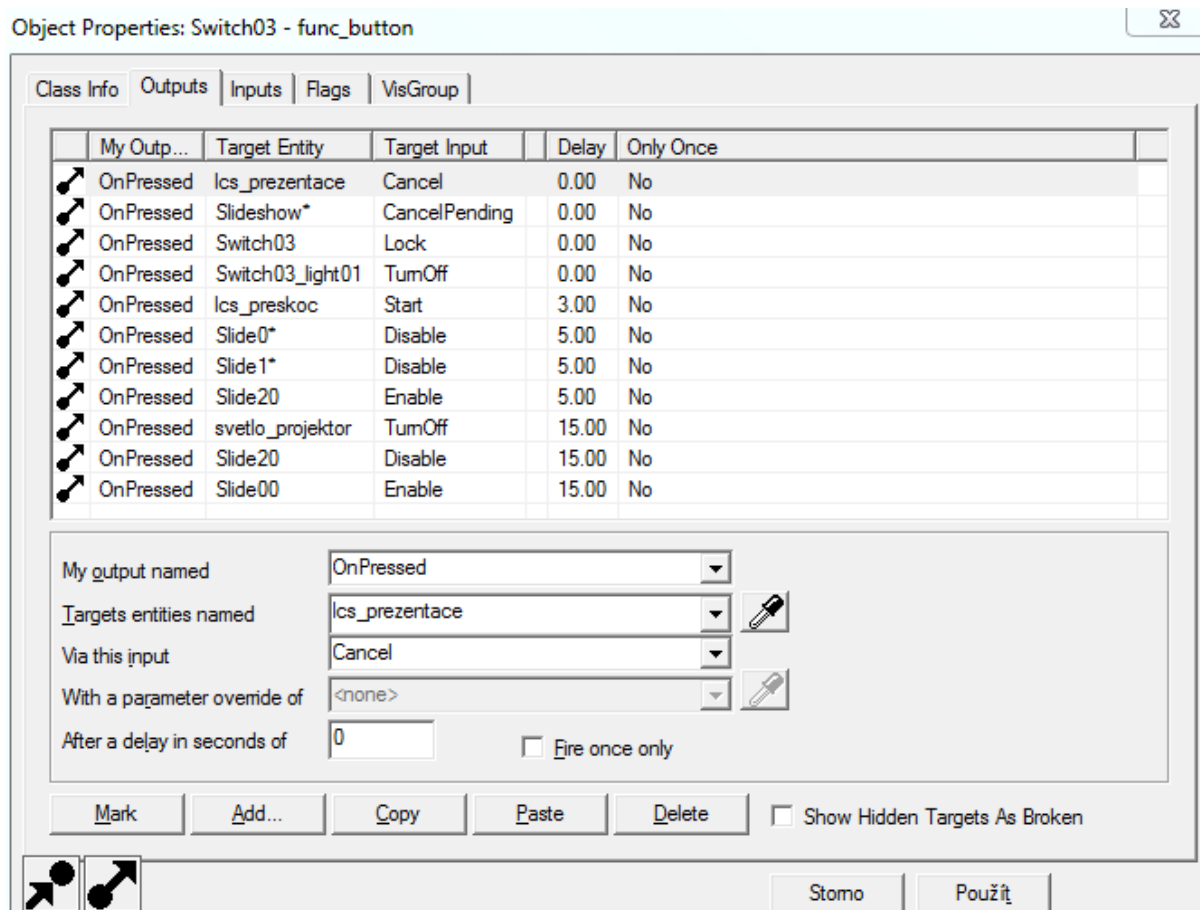
Světlo je řešeno pomocí entity *light\_spot* (světlo\_projektor). Ve vlastnostech této entity se nastaví jméno (Name), světelný efekt (Appearance) a úhel světelného kuželu (Outer angle). Dopad světla do roviny projekčního plátna je vidět na obrázku 6.1.



Obr. 6.1 - ukázka prezentační místnosti (VHE)

Po spuštění prezentace nás přivítá herec „prezentace“ (male\_08.mdl), který zájemce seznámí se základním konceptem výroby. Jednotlivé informace jsou doprovázeny načasovanými gesty a občasnou změnou pozice. V případě nedostatku času je možné tuto prezentaci přeskočit nebo kdykoliv ukončit stiskem entity *func\_button* (Switch03). Nastavení výstupů tlačítka je patrné z obrázku 6.2. Toto tlačítko ukončí právě přehrávanou scénu, vypne světlo od projektoru, zobrazí poslední slide na plátně a spustí scénu *lcs\_preskoc* (preskoc.vcd). Následně je divák vyzván k následování do výrobních prostor. Přesun postavy je řešen opět pomocí entit *path\_track* (path\_nahalu\_track\_1 až 5) s tím, že před každé dveře je přidán spouštěč (trigger\_otevri\_1 až 3), který včas otevře dveře. V nastavení v záložce Flags je třeba označit možnost NPCs, aby se dveře otvíraly. Před vstupními dveřmi na halu projdou v zorném poli hráče dvě postavy třídy *npc\_kleiner* (zena\_1 a zena\_2) do šaten. Tyto ženy mají ve vlastnostech v kolonce *Sleep State* nastaveno *Waiting for input*, což znamená, že čekají na místě, dokud je jiná entita neaktivuje. K aktivování dojde opět průchodem spouštěče (trigger\_zeny), který má nastavené výstupy s příkazem *Wake*. Příchodem do výrobních prostor aktuální scéna končí.





Obr. 6.2 – výstupy Switch03 (VHE)

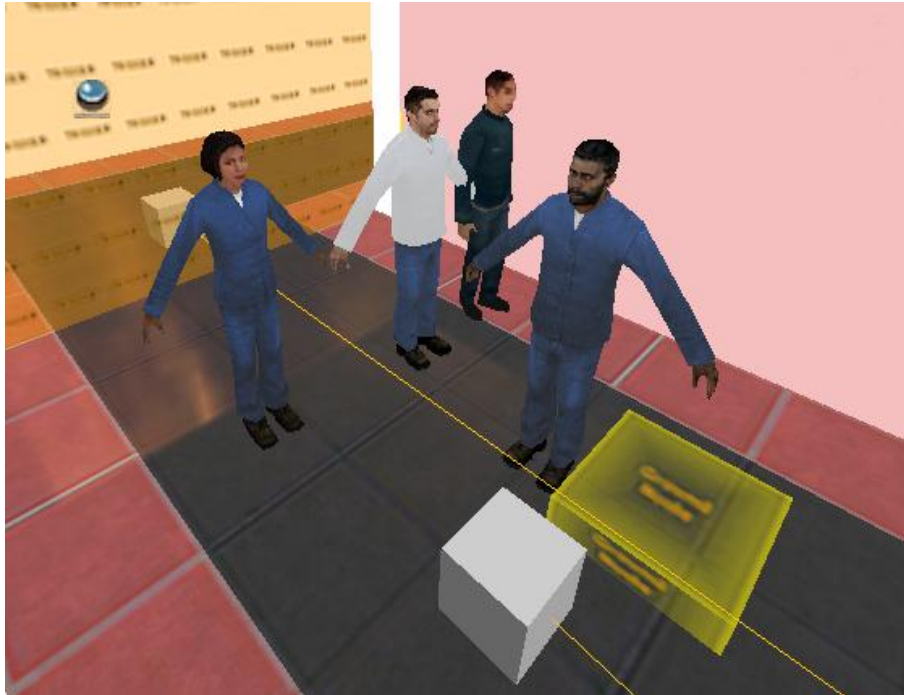
### 6.1.3 Výrobní hala

Po vstupu uživatele do výrobní haly se pomocí *trigger\_vyrobnihala* spustí choreografická scéna *lcs\_vyrobnihala* (vyrobnihala.vcd) a zároveň se aktivuje postava třídy *npc\_kleiner* (typ *pek\_vyroba*), která prochází celou výrobní linkou. Hercem ve scéně je postava „vyrobak“ (male\_02.mdl). Herec přijde blíž, opět hráče přivítá a vysvětlí, jak se dostane do vývojového oddělení. Dále poukáže na možnost vše si prohlédnout (včetně uchopení hotových RC modelů).

### 6.1.4 První patro

Poté, co vyjdeme po schodišti do prvního patra a projdeme spouštěcí entitou *trigger\_patro*, se spustí *lcs\_patro* (patro.vcd) a zároveň se aktivují 4 postavy, které opět projdou v zorném poli uživatele do vedlejší kanceláře (obrázek 6.3). Ve scéně *lcs\_patro* nám již známá „uvadecka“ ukáže dveře do kanceláře ředitele a do vývojového oddělení. Tato herečka opět reaguje na klávesu „E“ zopakováním své scény. Po vstupu do kanceláře ředitele nám sekretářka prostřednictvím *lcs\_ureditele* (ureditele.vcd) oznámí, že pan ředitel je zaneprázdňen. Tuto informaci si můžeme ověřit vstupem přímo k řediteli.





Obr. 6.3 - postavy čekající na vstup (VHE)

### 6.1.5 Vývojové oddělení

V tomto oddělení můžeme vstoupit do tří kanceláří (řízení výroby, technologie, konstrukce). Po vstupu do kanceláře řízení výroby se prostřednictvím *trigger\_vyroba* spustí scéna *lcs\_rizenivyroby* (*rizenivyroby.vcd*), ve které nás postava „vedoucí“ (*male\_06.mdl*) seznámí s náplní jejich práce. Na závěr nám představí vedoucího řízení výroby (*male\_08.mdl*), který vstane ze židle a zamává.

Po vstupu do kanceláře technologie projde uživatel entitou *trigger\_techologie*, která aktivuje scénu *lcs\_techologie* (*technologie.vcd*). Následuje příchod herce „technolog“ (*male\_02.mdl*), který vysvětlí, na čem pracují a představí hlavního technologa (*male\_01.mdl*). Hlavní technolog opět vstane a pokyne rukou.

Před dveřmi do kanceláře konstrukce jsme nuceni projít entitou *trigger\_chodba*, která aktivuje postavu „vchodbe“ (*kleiner.mdl*). Tato postava zamíří do kanceláře řízení výroby po nastavené trajektorii (*path\_vchodbe\_1* až *3*). Po otevření dveří do konstrukce se spustí scéna *lcs\_konstrukce* (*konstrukce.vcd*) pomocí *trigger\_konstrukce*. V této scéně jsme ústy postavy „konstrukter“ (*male\_04.mdl*) seznámeni s náplní práce a na závěr je nám představen hlavní konstruktér „malatny“ (*male\_06.mdl*), který vstane a zamává.

## 6.2 Zhodnocení

Celá ukázka je koncipována jako virtuální prohlídka továrny, ve které je uživatel vždy nasměrován na další místo. Uživatel se ale může sám rozhodnout, kam zamíří dříve a kam později. Spouštění scén pomocí entit třídy *trigger\_multiple* dává tedy jistou svobodu v rozhodování. Po překonání problémů s nástrojem Phoneme Tool (kapitola 5.3.1) se neobjevily žádné jiné nedostatky při tvorbě choreografických scén. Jediným dalším problémem byla poměrně velká časová náročnost při práci na delších scénách (*lcs\_prezentace*). V případě této scény bylo nutné sladit přehrávané zvukové stopy s běžící prezentací, což obnášelo neustálé zkoušení výsledku a opětovné změny.

### 6.3 Storyboard

Nyní se podívejme na vytvořenou virtuální prohlídku z pohledu uživatele.



Obr. 6.4 - ukázka virtuální prohlídky

- a) Externí pohled na výrobní podnik. Je možné prohlédnout parkoviště, popř. třeba vylézt na střechu. Zamíříme však dovnitř,
- b) kde budeme přivítání slovy: „Dobrý den, jste očekáván.“ Slečna nás dovede do prezentační místnosti, kde se nás ujme prezentátor,
- c) který nás během deseti minut seznámí se základní logikou systému výroby v podniku. Tuto prezentaci je možné přeskočit.
- d) Po ukončení prezentace jsme dovedeni do výrobní haly, kde je možné si prohlédnout každou dílčí část výroby.
- e) V administrativní části budovy jsme pak seznámeni v každé kanceláři s jednotlivými dílčími pracovišti,
- f) odkud je možno prohlédnout ještě jednu výrobu z ptačí perspektivy.

## 7 Tvorba výukového e-booku

Po dokončení popisu tvorby v aplikaci Face Poser, který je součástí této práce, bylo nutné vytvořené návody zpracovat do již existující publikace Polcar, J., Hořejší, P. a kolektiv, *Tvorba virtuálních prostředí v Source SDK*, e-book, ZČU 2012. Tento e-book je určen pro posluchače předmětu KPV/DPVR a popisuje práci se SW balíčkem Source SDK od instalace až po pokročilé funkce. V současné době je tedy tento e-book doplněn o aplikaci virtuálních postav do výrobního prostředí. Celý e-book je přiložen k této práci na CD.

Nová kapitola Face Poser popisuje tvorbu v této aplikaci od prvního spuštění, přes základní ovládání a vytvoření zvukové stopy, až po implementaci vytvořených scén do virtuálního prostředí pomocí VHE. Jednotlivé části víceméně korespondují s kapitolou 5 *Návody pro práci ve vybraném nástroji* této práce a jsou vhodně doplněny o praktické příklady.

- Face Poser
  - Úvod
  - Adresářová struktura
  - Ovládání
  - Vytvoření rámce scény
  - Načtení modelu postavy
  - Příprava a vložení zvukové stopy
  - Phoneme Tool
  - Vkládání a úprava gest
  - Animace obličeje
  - Implementace scény pomocí VHE
  - Příklad: První scéna
  - Orientace a pohyb v prostoru
  - Interakce
  - Case study: Prezentační místnost

## 8 Závěr

Cílem této práce bylo zmapování odborné literatury a internetových zdrojů na téma virtuální lidé a následná tvorba návodů k vybranému softwarovému nástroji. Byla vytvořena i praktická případová studie, která prokázala možnosti a vhodnost nasazení VH v praxi. Vytvořená simulace byla využita pro propagaci fakulty a bude využita v rámci dnů vědy na ulici 2013. Na závěr je možno říci, že i přes drobné problémy se stabilitou programu, si lze práci v softwaru Face Poser poměrně rychle osvojit a dosáhnout tak značného oživení a zvýšení interaktivity virtuálního prostředí. Velkým přínosem je, dle mého názoru, možnost takto významně snadněji zapůsobit na uživatele a prostředí lépe prezentovat bez nutnosti vysvětlování. Takto byly prakticky ověřeny poznatky o možnostech VH, které jsou popsány v úvodní části. Návodů jsou určeny pro výuku problematiky implementace a ovládání virtuálních lidí, ale využití tohoto softwaru je mnohem širší. Pro názornost lze uvést například prezentace projektů, interaktivní prohlídka plánované výstavby, instruktážní videa pro různé situace atp. Počty různých prvků využitých v praktické ukázce jsou uvedeny v tabulce 8.1. V této bakalářské práci byly použity výsledky z projektu OP VK č. CZ.1.07/2.3.00/09.0163.

Postav	20
Zvukových stop	50
Zvuku celkem v sekundách	365
Choreografických scén	15
Gest a postojů	120
Událostí <i>flex</i>	14
Událostí <i>lookat</i>	30
Událostí <i>faceto</i>	31
Událostí <i>moveto</i>	29

Tabulka 8.1 - statistika praktické ukázky

## Zdroje

- [1] Gutiérrez, M., Vexo, F., Thalmann, D.: *Stepping into Virtual Reality*, 2008, Springer-Verlag London, ISBN 978-1-84800-117-6
- [2] Magnenat-Thalmann, M., Thalmann, D.: *Handbook of Virtual Humans*, 2004, Wiley-IEEE Press, ISBN 0-470-02316-3
- [3] Digitální továrna [2012-10-24] Dostupné z: < <http://www.t-systems.cz/tsip/cs/606120/home/odvetvi/automobilovy-prumysl/digitalni-tovarna> >
- [4] Virtuální simulace výroby aneb Digitální továrna [2012-10-24] Dostupné z: < <http://www.systemonline.cz/rizeni-vyroby/virtualni-simulace-vyroby-aneb-digitalni-tovarna.htm> >
- [5] Coquillart, S. Brunnett, G., Welch, G.: *Virtual Realities*, 2011, Springer-Verlag Wien, ISBN 978-3-211-99177-0
- [6] Skeletal Animation [2012-10-29] Dostupné z :<[http://en.wikipedia.org/wiki/Skeletal\\_animation](http://en.wikipedia.org/wiki/Skeletal_animation)>
- [7] Görner, T., Hořejší, P., Kurkin, O., VYZTYMDP : Virtuální realita : úvodní úroveň, e-book, SmartMotion 2012, Plzeň, ISBN 978-80-87539-07
- [8] Character Motion Systems [2012-10-29] Dostupné z : <[http://www.siggraph.org/education/materials/HyperGraph/animation/character\\_animation/motion\\_capture/motion\\_optical.htm#Skeletal%20Animation](http://www.siggraph.org/education/materials/HyperGraph/animation/character_animation/motion_capture/motion_optical.htm#Skeletal%20Animation)>
- [9] Hořejší, P., Görner, T., Kurkin, O., VYZTYMDP : Virtuální realita : základní úroveň, e-book, SmartMotion 2012, Plzeň, ISBN 978-80-87539-07
- [10] Motion Capture Technology [2012-10-29] Dostupné z : <<http://freemotionfiles.blogspot.cz/2009/05/motion-capture-technology.html>>
- [11] Motion capture – řešení pro zrychlení animace [2012-11-25] Dostupné z: <[http://www.spafi.org/index.php?option=com\\_content&view=article&id=457%3Amotion-capture--solution-to-boost-up-animation&catid=37%3Ageneral-animation&Itemid=60&lang=cs](http://www.spafi.org/index.php?option=com_content&view=article&id=457%3Amotion-capture--solution-to-boost-up-animation&catid=37%3Ageneral-animation&Itemid=60&lang=cs)>
- [12] Skeleton-based rigid skinning for character animation [2012-10-17] Dostupné z: <[http://www.academia.edu/188921/Skeleton-based\\_Rigid\\_Skinning\\_for\\_Character\\_Animation](http://www.academia.edu/188921/Skeleton-based_Rigid_Skinning_for_Character_Animation)>
- [13] Autodesk 3ds Max [2012-10-26] Dostupné z: < <http://www.cadstudio.cz/3dsmax> >
- [14] T Systems [2012-10-25] Dostupné z: <<http://www.t-systems.cz/tsip/cs/606192/home/odvetvi/automobilovy-prumysl/digitalni-tovarna/catia-v5---human>>
- [15] DesignTech [2012-10-25] Dostupné z: <<http://www.designtech.cz/c/plm/ergonomicke-simulace-podnikovych-procesu.htm>>
- [16] Catia V5 – Human – řešení ergonomie práce [2012-10-25] Dostupné z: < <http://www.cad.cz/strojirenstvi/38-strojirenstvi/1855-catia-v5-human-reseni-ergonomie-prace.html> >
- [17] Virtual Reality in Medicine [2012-10-27] Dostupné z: <<http://www8.informatik.umu.se/~jwworth/medpage.html>>

- [18] Virtual Soldier Research [2012-10-26] Dostupné z :  
<<http://www.ccad.uiowa.edu/vsr/>>
- [19] Phoneme Tool [2013-3-27] Dostupné z:  
<[https://developer.valvesoftware.com/wiki/Phoneme\\_Tool](https://developer.valvesoftware.com/wiki/Phoneme_Tool)>
- [O1] Skeleton [2012-11-22] Dostupné z:  
<<https://cent.felk.cvut.cz/predmety/39PHA/data/prednasky/7/pha7skelet.pdf>>
- [O2] Linear blend skinning [2012-11-22] Dostupné z:  
<[http://dcgi.felk.cvut.cz/cgg/show\\_research.php?page=02](http://dcgi.felk.cvut.cz/cgg/show_research.php?page=02) >
- [O3] 3D body scanning [2012-11-22] Dostupné z: <<http://www.turbosquid.com/3d-models/realistic-body-3d-model/251252>>
- [O4] Inverzní kinematika [2012-11-22] Dostupné z : <  
[https://dip.felk.cvut.cz/browse/pdfcache/franc\\_2005dipl.pdf](https://dip.felk.cvut.cz/browse/pdfcache/franc_2005dipl.pdf)>
- [O5] Mechanical motion capture [2012-10-29]:  
<<http://freemotionfiles.blogspot.cz/2009/05/motion-capture-technology.html>>
- [O6] Optical motion capture [2012-10-29] Dostupné z:<  
<http://freemotionfiles.blogspot.cz/2009/05/motion-capture-technology.html>>
- [O7] Magnetic mocap: <<http://www.golf.com/photos/motion-picture/motion-capture-technology>>
- [O8] Centaur Concept [2012-11-22] Dostupné z: <<http://sketch-site.blogspot.cz/2010/07/centaur-concept.html>>
- [O9] Ananova [2012-10-22] Dostupné z :  
<<http://www.charamel.com/avatare/customized.html>>
- [O10] Tomb Raider Underworld [2012-10-23] Dostupné z:  
<<http://tombraderreborn.blog.cz/galerie/tomb-raider-underworld#>>
- [O11] The Sims 3 [2012-10-23] Dostupné z : <<http://blog.gamer20.com/2009/06/review-the-sims-3-pc/>>
- [O12] Tecnomatix Jack [2012-10-24] Dostupné z :  
<<http://www.deskeng.com/articles/aaaxns.htm>>
- [O13] Delmia V5 – Human [2012-10-24] Dostupné z: <http://www.cad.cz/strojirenstvi/38-strojirenstvi/1855-catia-v5-human-reseni-ergonomie-prace.html>
- [O14] Techniques of Medical VR [2012-10-25] Dostupné z:  
<<http://www8.informatik.umu.se/~jwworth/2Techniques#2.1>>
- [O15] Virtual Soldier Research brochure [2012-10-25] Dostupné z:  
<[http://www.ccad.uiowa.edu/vsr/index.php/download\\_file/-/view/248/](http://www.ccad.uiowa.edu/vsr/index.php/download_file/-/view/248/)>
- [O16] Musculoskeletal Model [2012-10-25] Dostupné z:<<http://www.ccad.uiowa.edu/vsr/research/musculoskeletal-model/>>
- [O17] Reflex reactions to a ball: Gutiérrez, M., Vexo, F., Thalmann, D.: Stepping into Virtual Reality, 2008, Springer-Verlag London, ISBN 978-1-84800-117-6

## **Příloha č. 1**

**Skript game\_sounds\_manifest.txt**



game\_sounds\_manifest

```
{  
    "precache_file"          "scripts/game_sounds.txt"  
    "precache_file"          "scripts/game_sounds_ui.txt"  
    "precache_file"          "scripts/game_sounds_player.txt"  
  
    // Weapon sounds  
    "precache_file"          "scripts/game_sounds_weapons.txt"  
    "precache_file"          "scripts/game_sounds_weapons_ep2.txt"  
  
    // HL2 World, Ambient Generic, Items, Physics, Vehicles  
    "precache_file"          "scripts/game_sounds_world.txt"  
    "precache_file"          "scripts/game_sounds_ambient_generic.txt"  
    "precache_file"          "scripts/game_sounds_items.txt"  
    "precache_file"          "scripts/game_sounds_physics.txt"  
    "precache_file"          "scripts/game_sounds_vehicles.txt"  
  
    // E3 level sounds  
    "precache_file"          "scripts/level_sounds_e3_c17.txt"  
    "precache_file"          "scripts/level_sounds_e3_town.txt"  
    "precache_file"          "scripts/level_sounds_e3_bugbait.txt"  
  
    // Episodic content sounds  
    "precache_file"          "scripts/npc_sounds_alyx_episodic.txt"  
    "precache_file"          "scripts/npc_sounds_strider_episodic.txt"  
    "precache_file"          "scripts/npc_sounds_strider_episodic2.txt"  
    "precache_file"          "scripts/npc_sounds_turret_episodic.txt"  
    "precache_file"          "scripts/npc_sounds_soldier_episodic.txt"  
    "precache_file"          "scripts/npc_sounds_ministrider_episodic.txt"
```

"precache_file"	"scripts/npc_sounds_roller_episodic.txt"
"precache_file"	"scripts/npc_sounds_combine_ball_episodic.txt"
"precache_file"	"scripts/npc_sounds_citizen_episodic.txt"
"precache_file"	"scripts/npc_sounds_citizen_episodic2.txt"
"precache_file"	"scripts/npc_sounds_citizen_ep1.txt"
"precache_file"	"scripts/npc_sounds_zombine.txt"
"precache_file"	"scripts/npc_sounds_dog_episodic.txt"
"precache_file"	"scripts/npc_sounds_antlion_episodic.txt"
"precache_file"	"scripts/npc_sounds_antlionguard_episodic.txt"
"precache_file"	"scripts/npc_sounds_antlionguard_episodic2.txt"
"precache_file"	"scripts/npc_sounds_advisor.txt"
"precache_file"	"scripts/npc_sounds_advisor_episodic2.txt"
"precache_file"	"scripts/npc_sounds_hunter.txt"
"precache_file"	"scripts/npc_sounds_antlion_grub_episodic.txt"
"precache_file"	"scripts/npc_sounds_attackheli_episodic2.txt"
"precache_file"	"scripts/npc_sounds_fastzombie_episodic2.txt"
"precache_file"	"scripts/game_sounds_addendum_ep2.txt"
"precache_file"	"scripts/level_sounds_music_episodic.txt"
"precache_file"	"scripts/level_sounds_music_episodic2.txt"
"precache_file"	"scripts/game_sounds_weapons_episodic.txt"
"precache_file"	"scripts/level_voices_episode_01.txt"
"precache_file"	"scripts/level_sounds_aftermath_episodic.txt"
"precache_file"	"scripts/level_sounds_outland_episodic.txt"
"precache_file"	"scripts/level_sounds_c17_02a.txt"
"precache_file"	"scripts/level_voices_episode_02.txt"
"precache_file"	"scripts/game_sounds_vehicles_ep2.txt"

// EP2 content sounds

"precache\_file" "scripts/game\_sounds\_physics\_ep2.txt"

// Game level sounds

"precache\_file" "scripts/level\_sounds\_eli\_lab.txt"

"precache\_file" "scripts/level\_sounds\_trainyard.txt"

"precache\_file" "scripts/level\_sounds\_k\_lab.txt"

"precache\_file" "scripts/level\_sounds\_k\_lab2.txt"

"precache\_file" "scripts/level\_sounds\_coast.txt"

"precache\_file" "scripts/level\_sounds\_novapropekt.txt"

"precache\_file" "scripts/level\_sounds\_streetwar.txt"

"precache\_file" "scripts/level\_sounds\_streetwar2.txt"

"precache\_file" "scripts/level\_sounds\_breencast.txt"

"precache\_file" "scripts/level\_sounds\_citadel.txt"

"precache\_file" "scripts/level\_sounds\_canals.txt"

"precache\_file" "scripts/level\_sounds\_ravenholm.txt"

"precache\_file" "scripts/level\_sounds\_ravenholm2.txt"

"precache\_file" "scripts/level\_sounds\_canals2.txt"

//Music Tracks

"precache\_file" "scripts/level\_sounds\_music.txt"

// NPC Sounds

"precache\_file" "scripts/npc\_sounds\_eli.txt"

"precache\_file" "scripts/npc\_sounds\_alix.txt"

"precache\_file" "scripts/npc\_sounds\_dog.txt"

"precache\_file" "scripts/npc\_sounds\_citizen.txt"

"precache\_file" "scripts/npc\_sounds\_barney.txt"

"precache_file"	"scripts/npc_sounds_soldier.txt"
"precache_file"	"scripts/npc_sounds_strider.txt"
"precache_file"	"scripts/npc_sounds_zombie.txt"
"precache_file"	"scripts/npc_sounds_vortigaunt.txt"
"precache_file"	"scripts/npc_sounds_turret.txt"
"precache_file"	"scripts/npc_sounds_scanner.txt"
"precache_file"	"scripts/npc_sounds_rollermine.txt"
"precache_file"	"scripts/npc_sounds_poisonzombie.txt"
"precache_file"	"scripts/npc_sounds_metropolice.txt"
"precache_file"	"scripts/npc_sounds_combinecamera.txt"
"precache_file"	"scripts/npc_sounds_manhack.txt"
"precache_file"	"scripts/npc_sounds_ichthyosaur.txt"
"precache_file"	"scripts/npc_sounds_blackheadcrab.txt"
"precache_file"	"scripts/npc_sounds_fastheadcrab.txt"
"precache_file"	"scripts/npc_sounds_headcrab.txt"
"precache_file"	"scripts/npc_sounds_fastzombie.txt"
"precache_file"	"scripts/npc_sounds_birds.txt"
"precache_file"	"scripts/npc_sounds_gunship.txt"
"precache_file"	"scripts/npc_sounds_dropship.txt"
"precache_file"	"scripts/npc_sounds_barnacle.txt"
"precache_file"	"scripts/npc_sounds_attackheli.txt"
"precache_file"	"scripts/npc_sounds_antlionguard.txt"
"precache_file"	"scripts/npc_sounds_antlion.txt"
"precache_file"	"scripts/npc_sounds_env_headcrabcanister.txt"
"precache_file"	"scripts/npc_sounds_combine_ball.txt"
"precache_file"	"scripts/npc_sounds_combine_mine.txt"
"precache_file"	"scripts/npc_sounds_sniper.txt"
"precache_file"	"scripts/npc_sounds_stalker.txt"
"precache_file"	"scripts/npc_sounds_gman.txt"

```
"precache_file"          "scripts/npc_sounds_combine_cannon.txt"
"precache_file"          "scripts/npc_sounds_alyx_episodic2.txt"
"precache_file"          "scripts/npc_sounds_turret_episodic2.txt"

//Moje zvuky
// Game level sounds
"precache_file"          "scripts/level_sounds_demonstr.txt"
"precache_file"          "scripts/level_sounds_prezentace.txt"
"precache_file"          "scripts/level_sounds_konstrukce.txt"
"precache_file"          "scripts/level_sounds_patro.txt"
"precache_file"          "scripts/level_sounds_rizeni.txt"
"precache_file"          "scripts/level_sounds_technologie.txt"
"precache_file"          "scripts/level_sounds_vyroba.txt"
"precache_file"          "scripts/level_sounds_ureditele.txt"
}
```

## **Příloha č. 2**

**Skript level\_sounds\_konstrukce.txt**

```
"konstrukce_1"  
{  
    "channel"          "CHAN_VOICE"  
    "volume"          "0.900"  
    "pitch"           "PITCH_NORM"  
    "soundlevel"     "SNDLVL_TALKING"  
    "wave"            "*vo/konstrukce/konstrukce_1.wav"  
}  
"konstrukce_2"  
{  
    "channel"          "CHAN_VOICE"  
    "volume"          "0.900"  
    "pitch"           "PITCH_NORM"  
    "soundlevel"     "SNDLVL_TALKING"  
    "wave"            "*vo/konstrukce/konstrukce_2.wav"  
}
```