

UNIVERZITA PARDUBICE

Fakulta elektrotechniky a informatiky

Propagace FEI a podpora výuky pomocí agentů
společnosti Microsoft

Jan Stárka

Bakalářská práce

2010

Univerzita Pardubice
Fakulta elektrotechniky a informatiky
Akademický rok: 2009/2010

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Jan STÁRKA**
Osobní číslo: **I07787**
Studijní program: **B2646 Informační technologie**
Studijní obor: **Informační technologie**
Název tématu: **Propagace FEI a podpora výuky pomocí agentů společnosti Microsoft**
Zadávající katedra: **Katedra informačních technologií**

Z á s a d y p r o v y p r a c o v á n í :

V teoretické části bakalářské práce je nutné provést přehled problematiky a navrhnout možné řešení. Zvláště zmapování všech nástrojů a technik pro zvládnutí animačních sekvencí, bublinových dialogů, zvukových výstupů, časové koordinace více agentů v interakci programování ovládání agentů.

Implementační část bude obsahovat

* Vytvoření nového agenta reprezentující FEI, tento agent bude mít reprezentativní vzhled znázorňující zaměření fakulty. U tohoto nového agenta bude vytvořeno cca 12 animačních sekvencí, tyto animace budou navrženy tak, aby podporovaly nebo zvýrazňovaly přednášené téma (např. poukazování na nadpisy, příchod a odchod z obrazovky apod.).

* Řešení problematiky dabingu agentů.

* Vytvoření ukázkové prezentace, kterou bude řídit nadabovaný agent pro účely propagace FEI na středních školách.

* Vytvoření ukázkové přednášky, kterou bude agent řídit (animační sekvence, bublinová dialogová okna). Prezentování přednášek bude řešeno v Microsoft Office PowerPoint a ovládání agenta v prostředí jazyka C.

* Výsledkem práce bude komentovaný zdrojový kód ovládající nově vytvořeného agenta a podpůrné aplikace (např. Windows Media Player pro přehrávání zvuků) v jazyce C spolu s programátorskou příručkou.

Rozsah grafických prací:

Rozsah pracovní zprávy:

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

1. NAGEL, Christian. C 2008 Programujeme profesionálně, Brno, Computer Press, 2009
2. Microsoft Agent home page [online]. Dostupný z WWW: <<http://www.microsoft.com/products/msagent/main.aspx>>
3. Microsoft Developer Network - Microsoft Agent [online]. Dostupný z WWW: <[http://msdn.microsoft.com/en-us/library/ms695784\(VS.85\).aspx](http://msdn.microsoft.com/en-us/library/ms695784(VS.85).aspx)>
4. Microsoft Developer Network - Assistant Object [online]. Dostupný z WWW: <http://msdn.microsoft.com/en-us/library/aa432034.aspx>
5. Microsoft Developer Network - Using the Microsoft Agent Character Editor [online]. Dostupný z WWW: <[http://msdn.microsoft.com/en-us/library/ms695797\(VS.85\).aspx](http://msdn.microsoft.com/en-us/library/ms695797(VS.85).aspx)>
6. MASH - Microsoft Agent Scripting Helper [online]. Dostupný z WWW: <<http://www.bellcraft.com/mash/>>
7. Microsoft Agent Ring [online]. Dostupný z WWW: <<http://www.msagentring.org/>>

Vedoucí bakalářské práce:

Ing. Zbyněk Kopecký

Katedra informačních technologií

Datum zadání bakalářské práce:

15. ledna 2010

Termín odevzdání bakalářské práce:

14. května 2010



prof. Ing. Simeon Karamazov, Dr.
děkan



L.S.



Ing. Lukáš Čegan, Ph.D.
vedoucí katedry

V Pardubicích dne 31. března 2010

Prohlášení autora

Prohlašuji, že jsem tuto práci vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně.

V Pardubicích dne 20. 6. 2010

Jan Stárka

Anotace

Bakalářská práce se zabývá animací a dubbingem postavy, která bude využívána pro vedení prezentace určené pro střední školy. Prezentace využívá dubbované postavy k podpoře přednášejícího. Aplikace je navržena v prostředí Visual C# a využívá podpůrné aplikace „PowerPoint 2003“ a „Windows Media Player“.

Klíčová slova

animace, snímek, postava, Microsoft Agent, Visual C#, prezentace

Title

FEI promotion and support of teaching using Microsoft Agents.

Annotation

The diploma paper deals with animation and dubbing of character, which will be used for leading the presentation for high schools. As a support for a lecturer are used dubbed character. The application is designed in Visual C# and uses supporting application „Power Point 2003“ and „Windows Media Player“.

Keywords

animation, image, character, Microsoft Agent, Visual C#, presentation

Obsah

Seznam zkratk	8
Seznam obrázků	9
Seznam tabulek	9
1 Úvod	10
2 Základní informace o Microsoft Agentech	11
2.1 Účel.....	11
2.2 Použití.....	11
2.3 Vývojové prostředí	11
2.4 Systémové požadavky	12
3 Návrh postav a Microsoft Agent Control 2.0	12
3.1 Postava.....	12
3.2 Animace.....	14
3.2.1 Snímky.....	14
3.2.2 Návrh obrázků	14
3.2.3 Velikost snímku.....	15
3.2.4 Časová délka snímku	16
3.2.5 Snímkový přechod.....	16
3.2.6 Animace pohybu úst.....	17
3.3 Stavy Agenty	18
3.3.1 Stavy „Hearing“ a „Listening“	19
3.3.2 Stav „Gesturing“	19
3.3.3 Stavy „Showing“ a „Hiding“	19
3.3.4 Stav „Idling“	20
3.3.5 Stav „Speaking“	20
3.3.6 Stav „Moving“	20
3.3.7 Standardní animační sada	21
3.4 Zásady animace	22
3.4.1 Technika „Squash and Stretch“	22
3.4.2 Očekávání	23
3.4.3 Načasování.....	23
3.4.4 Inscenace	23

3.4.5	„Slow In-and-Out“	23
3.4.6	Vektor pohybu	23
3.4.7	Zveličování	23
3.4.8	Sekundární akce.....	24
3.4.9	Solidní kresba	24
3.4.10	Apelování	24
3.5	Uživatelské rozhraní	24
3.5.1	Ikona v panelu nástrojů.....	24
3.5.2	Textové balóny	25
3.5.3	Software pro rozpoznání hlasu	25
4	Postup animace v praktické části	26
4.1	Tvorba snímků.....	26
4.2	Řazení snímků pomocí aplikace Microsoft Agent Character Editor.....	28
4.2.1	Aplikace „Microsoft Agent Character Editor“	28
4.3	Animační sekvence.....	29
4.3.1	„Start“	29
4.3.2	„Konec“	29
4.3.3	„Mrknuti“	29
4.3.4	„Neaktivni1“	30
4.3.5	„Neaktivni2“	30
4.3.6	„Vystraha“	30
4.3.7	„Naslouchani“.....	30
4.3.8	„Mluveni“	31
4.3.9	„Pohyb“	31
4.3.10	„KonecPohybu“	31
4.3.11	„UkazaniNahoru“	31
4.3.12	„UkazaniDoprava“.....	31
4.3.13	„UkazaniDoleva“	31
4.3.14	„UkazaniDolu“	31
5	Použití postavy v aplikaci	33
5.1	Metody knihovny „Microsoft Agent Control 2.0“	33
5.2	Implementace pomocí „Visual C#“	34
5.2.1	Inicializace postavy	34

5.2.2	Pohyb po obrazovce	35
5.2.3	Přehrávání animací	35
5.2.4	Mluvení postavy	36
5.2.5	Synchronizace agentů	36
5.3	Popis vývoje praktické úlohy	38
5.3.1	Načtení a zobrazení Microsoft Agentů	38
5.3.2	Spuštění aplikace Microsoft PowerPoint.....	38
5.3.3	Použití Windows Media Player	39
5.4	První úloha praktické části – popis.....	39
5.5	Druhá úloha praktické části – popis	40
6	Závěr	41
	Literatura	42

Seznam zkratek

JPEG	Joint Photographic Experts Group
BMP	Microsoft Windows Bitmap
WAV	Waveform Audio Format
GIF	Graphics Interchange Format
PNG	Portable Graphics network
PC	Personal Computer

Seznam obrázků

Obrázek 1 - První návrh agenta reprezentující FEI nazvaný „Robi“	13
Obrázek 2 - Druhý návrh agenta reprezentující FEI nazvaný „Elektronka“	13
Obrázek 3 - Třetí návrh agenta reprezentující FEI nazvaný „C#“, je použit v práci.....	13
Obrázek 4 - Neutrální poloha Microsoft Agentu použitého v bakalářské práci	17
Obrázek 5 - Ukázka ikony Microsoft Agentu „Merlina“ na hlavním panelu systému Windows.....	25
Obrázek 6 - Ukázka balónového dialogu „Microsoft Agent 2.0“	25
Obrázek 7 - Ukázka negativního vlivu antialiasingu na animovanou postavu.....	26
Obrázek 8 - Ukázka polotovaru animované postavy „Sharp“	27
Obrázek 9 - Ukázka aplikace „Microsoft Agent Character Editor“	28
Obrázek 10 - Ukázka s animační sekvence „Vystraha“	30
Obrázek 11 – Ovládací panel první úlohy	39
Obrázek 12 – Ovládací panel druhé úlohy	40

Seznam tabulek

Tabulka 1 - Polohy úst v aplikaci „Microsoft Agent Character Editor“	17
Tabulka 2 - Stavy postav „Microsoft Agent 2.0“	18
Tabulka 3 - Standartní animace „Microsoft Agent 2.0“	21
Tabulka 4 - Animace přiřazené ke stavům postavy.....	22
Tabulka 5 - Tabulka animací postavy „ Csharp“	32

1 Úvod

Cílem této bakalářské práce je animace a dubbing postavy „Microsoft Agent“. Po té navrhnout aplikaci využívající animovanou postavu, která k dubbingu využívá „Windows Media Player“ a k přehrávání prezentace aplikaci „PowerPoint 2003“. Prezentace reprezentuje fakultu Elektrotechniky a Informatiky a je určena pro střední školy.

Druhá kapitola je zaměřena na seznámení s prostředím „Microsoft Agent 2.0“, jejím základním účelem a možným použitím.

Ve třetí kapitole je uvedeno jak efektivně postavu animovat, tak aby vypadala věrohodně. Kapitola dále obsahuje tři návrhy postavy a jejich animace. Je zde uveden způsob řešení animací, i za jakých situací jsou animace spouštěny. Tato kapitola je věnována technikám animace postav „Microsoft Agent 2.0“.

Čtvrtá kapitola popisuje průběh tvorby postavy „Microsoft Agent 2.0“. Jsou zde uvedeny problémy, se kterými jsem se v průběhu animace setkal a jejich následné řešení, počínaje tvorbou snímků až po jejich finální sestavování. V závěru kapitoly je seznam animačních sekvencí, které postava obsahuje.

Pátá kapitola popisuje implementaci postavy „Microsoft Agent“ pomocí programovacího prostředí „Visual C#“. Je zde uveden přehled použitých knihoven a metod včetně jejich implementace ve zdrojovém kódu. V závěru kapitoly jsou popsány vytvořené aplikace.

V Poslední kapitole je uveden výsledek praktické části a zhodnocuje úspěchy či neúspěchy během její tvorby.

2 Základní informace o Microsoft Agentech

2.1 Účel

„Microsoft Agent 2.0“ poskytuje technologii, která tvoří nové komunikační rozhraní pro aplikace a webové stránky. To umožňuje silná animační způsobilost, interaktivita a přizpůsobivost.[3]

Microsoft Agent je technologie poskytující základ pro přirozenější způsoby komunikace lidí se svým počítačem. Je to sada softwarových služeb, které umožní vývojáři včlenění interaktivní animované postavy do jejich aplikací a webových stránek. Tyto postavy mohou mluvit přes speciální software, převádějící text na mluvené slovo, nebo pomocí nahraného audio záznamu a dokonce přijmout mluvené hlasové příkazy. Microsoft Agent posiluje vývojáře pro rozšířené uživatelské rozhraní mimo konvenční myš a klávesnici, které dnes převládají.[3]

Vylepšení aplikací a webových stránek interaktivní osobou polidšťuje interakci mezi uživateli a jejich osobními počítači.[3]

2.2 Použití

Díky velké podpoře nejrůznějších vývojových prostředí lze „Microsoft Agent 2.0“ prakticky použít v jakékoliv aplikaci nebo funkci, kde je zapotřebí komunikace s osobami obsluhujícími osobní počítač. Níže jsou uvedeny některé možnosti případného využití pomocníka:[3]

- Přivítání a prohlídka při prvním spuštění PC, aplikace nebo webové stránky
- Přátelský tutoriál vedoucí přes úkoly využívající rozhodovací strom s pokyny krok za krokem.
- Animovaný pomocník oznamující příchod nového e-mailu včetně nabídky jeho otevření.
- Plnění různých úkolů, jako vyhledání klíčových slov na webu a následné přečtení výsledků apod.

2.3 Vývojové prostředí

„Microsoft Agent 2.0“ je určen především pro vývojáře, kteří používají jazyky nebo vývojová prostředí podporující ovládací rozhraní „COM“ nebo „Microsoft ActiveX“. Mezi která patří:[3]

- Microsoft Visual Studio (Visual C++, Visual Basic, Visual C#)
- Microsoft Office (Visual Basic pro aplikace)
- Microsoft Internet Explorer (Visual Basic Scripting Edition nebo Microsoft Jscript)
- Microsoft Windows Script Host (ActiveX Scripting Language)
- Další aplikace a prostředí podporující ovládací prostředí „COM“ nebo „ActiveX“.

2.4 Systémové požadavky

- Microsoft Windows 95, Windows 98, Windows Me, Windows NT 4.0 (x86), Windows 2000, Windows Xp, Windows Vista.
- Internet Explorer verze 3.02 nebo novější
- Pentium 100 MHz PC (nebo vyšší)
- 16 MB RAM

3 Návrh postav a Microsoft Agent Control 2.0

3.1 Postava

Lidé rádi komunikují s lidmi. „Microsoft Agent“ umožňuje využít tento aspekt interakce pomocí animovaných postav. Uživatelé očekávají, že postava splňuje stejná sociální pravidla. Ne však fyzická, která používají při interakci s jinými lidmi, přičemž si uvědomují, že animovaná postava je syntaktická. Do té míry můžeme vytvořit postavy splňující jejich očekávání. Z toho vyplývá důležitost návrhu postavy pro její úspěch.[3]

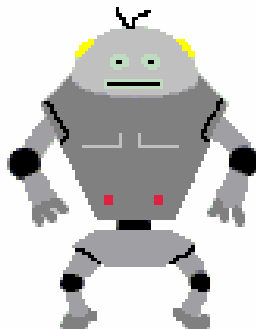
Při návrhu postavy je důležité zvážit profil cílové skupiny uživatelů i jaké úkoly bude provádět v jednotlivých animacích. V mém případě jsem vybral C#. Tento symbol dobře reprezentuje Fakultu elektrotechniky a informatiky, jako jedno z vývojových prostředí používané v některých předmětech naší fakulty. Například pro vybraní návrhu nějaké bezpečnostní aplikace by se hodil pes. Stejně tak výběr jednotlivých animačních sekvencí je důležitý z hlediska kontextu s podávanou informací.[3]

Další důležitý aspekt je výběr osobnosti animované postavy např. dominantní, submisivní, emociální, vyhraněné nebo sofistikované. Návrh osobnosti lze přenechat uživateli snadným nastavením nebo dle reakce na pokyn uživatele. V tomto případě návrhu lze počítat s vyšší náročností tvorby postavy.[3]

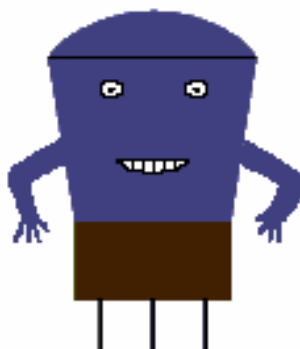
Co se týče osobnosti, důležitým aspektem je i výběr jména. Postava jménem „Max“ působí úplně jinak než postava jmenující se „Míša“.

Při výběru hlasového výstupu je třeba uvážit, zda použít hlas syntetický, nebo záznam zvuku ve formátu „WMA“ a podobných. Toto rozhodnutí může záviset na typu postavy a jazyku, který postava používá (anglický jazyk, český jazyk atd.). Syntetizovaný hlas má výhodu schopnosti říci téměř cokoliv rychle a snadno. Stačí jen dodat text a postava už bude mluvit sama. Nevýhodou syntaktické mluvy je někdy nejasnost a nepodobnost lidské řeči. V případě anglického jazyka lze snadno sehnat software pro převádění textu na mluvené slovo, ale v ostatních jazycích už to není tak jednoduché, což je velká nevýhoda. Pokud postava nepotřebuje značný rozsah vět k mluvě, lze bez větších problémů postavu namluvit na záznam nějakého zvukového formátu například „WMA“.

Také lze nastavit atributy balónového dialogu pro text. Tyto atributy jsou font písma, barva písma, barva rámečku i barva plochy bubliny, i přesto mají uživatelé možnost měnit nastavení.



Obrázek 1 - První návrh agenta reprezentující FEI nazvaný „Robi“



Obrázek 2 - Druhý návrh agenta reprezentující FEI nazvaný „Elektronka“



Obrázek 3 - Třetí návrh agenta reprezentující FEI nazvaný „C#“, je použit v práci.

Z výše uvedených návrhů jsem pro svoji práci vybral Microsoft agenta „Csharp“, kvůli jeho jednoduchosti, ale zároveň výstižnosti naší fakulty. Animace jsou jednoduché a efektivní. Nechal jsem se inspirovat postavou „Svorka“ z „Microsoft Excel“.

3.2 Animace

Animace postavy odráží její pohlaví, věk, osobnost a chování. Variabilita postavy závisí na potřebě komunikace mezi uživatelem a postavou. Pro méně rozsáhlé aplikace není třeba vyšších dovedností postavy.[3]

Stejně jako tradiční animace, tak i digitální, se skládá ze série mírně odlišných obrázků a při jejich postupném zobrazení pro lidské oko nastává dojem plynulého pohybu. Vytváření vysoce kvalitních snímků animace vyžaduje kvalifikovaného animátora. Dobře vystižený styl a prezentace postavy zaručí jeho efektivitu i bez něj. Dvourozměrné postavy s jednotlivými tvary a rysy mohou být někdy stejně i více efektivní, než kvalitně animované postavy. Není nutné vytvořit realistický obraz. Mnoho populárních kreslených postavček není realistických ve své prezentaci, ovšem přesto jsou efektivní díky animátorově zprostředkování akcí a emocí.[3]

3.2.1 Snímky

Každá animace vytvořená pro postavu „Microsoft Agent“ se skládá z časové sekvence snímků. Každý snímek v animaci je tvořen z jednoho nebo více bitmapových obrázků. Velikosti obrázků mohou být dle potřeby rozmanité.[3]

Animace drobností, jako například mrkající oko nebo pohyb prstů, mohou být zahrnuty v dalších obrázcích pro snímek. Vhodné je překrývání několika vrstev, které tvoří celkovou kompozici a liší se v jednotlivých vrstvách. Tato technika umožňuje opakované použití detailních obrázků (např. oči, pusa, ruce atd.) ve více snímcích. Například u mé postavy jsem použil v různých vrstvách kombinace oči. Jednou jsou otočeny doprava, příště doleva atd. Všechny možnosti jsou uloženy a připraveny k opětovnému použití. Tato technika nejen, že ušetří čas, ale také zaručí u postavy jistou symetrii.[3]

Animace umožňuje libovolný počet snímků. Obvyklý počet snímků je 24 s dobou trvání maximálně 6 sekund. Tento umírněný časový úsek zajistí postavě plynulý pohyb pro oko uživatele. Kromě toho vyšší počet snímků způsobuje větší velikost souboru. Tato problematika působila problémy v době, kdy byl internet ještě pomalý a u webové stránky by animace mohla probíhat neplynule. U animací s vyšší robustností pohybu se však zvýšenému počtu snímků nevyhneme.[3]

3.2.2 Návrh obrázků

Pro tvorbu obrázků animace lze použít jakýkoliv grafický nástroj za předpokladu ukládání obrázků v bitmapě Windows (.BMP) formátu. Pokud jsou obrázky vytvořeny, použijí se v aplikaci „Microsoft Agent Charakter Editor“. Tento postup bude rozveden na konci této kapitoly.

Obrázky musí být navrženy v 256 barevné paletě, ovšem 20 barev je standardně rezervováno systémem Windows. Jejich běžná pozice na paletě se nachází v prvních 10-ti a posledních 10-ti pozicích. To znamená, že pro animovanou postavu může být použito 236 barev z uvedené palety. Při definici své palety budou barvy přepsány barvami systémovými, které se dané barvě nejvíce podobají, tedy pokud daná barva není barvou systémovou. Čím širší paletu má postava nadefinovaná, tím je větší pravděpodobnost předefinování dané palety.[3]

Další problém s vyšším počtem barev na paletě spočívá ve velikosti obrázku. Čím je paleta širší, tím roste velikost souboru. Počet barev a frekvence obměny rozhodují, jak vysoce bude soubor komprimován. Například dvourozměrná postava používající úzkou škálu barev bude komprimovaná mnohem lépe, než postava třírozměrná s mnoha odstíny barev.

Pro celý soubor „Microsoft Agent“ musí být použity snímky se stejnou paletou barev. Tedy před animováním je nutné dopředu dobře uvážit výběr palety.

11. místo v paletě je definováno standardně jako průhledná, nebo-li „alfa“ barva. Také lze barvu nastavit pomocí „Microsoft Agent charakter Editor“. Animační služby „Microsoft Agent“ poskytují transparentnost barvy pro námi vybraný odstín, v případě chceme-li transparentnost použít.

Je také zapotřebí pečlivě zvážit tvar postavy, protože to může ovlivnit výkon animace. Pro zobrazení postavy vytvářejí animační služby oblastní okna na celé ploše obrázku. Malé nepravidelné plochy často vyžadují rozsáhlá data, což způsobuje u starších PC snížení animačního výkonu postavy. Proto je nutné se vyhnout mezerám nebo jednotlivým pixelům uprostřed plochy s odlišným odstínem.

Další problém, který mi způsobil nemalé potíže je „antialiasing“ vyhlazující vnější okraje postavy. Vzhledem k tomu, že se postava může ocitnout na vrcholu jakékoliv barvy, vyhlazené vnější okraje postavy mohou vykreslit světlé odstíny. „Antialiasing“ se však smí používat na vnitřních hranách. Tam se už tento problém neprojeví.

3.2.3 Velikost snímku

Velikost snímků by typicky neměla být větší než 128 x 128 pixelů. Přestože postavy mohou mít větší nebo menší rozměr snímků, „Microsoft Agent Character Editor“ používá pro zobrazení postavy toto rozlišení jako standard. V mém případě postavu využívám jako doprovod prezentace. V tomto případě je vhodné, aby postava měla větší rozměry kvůli lepší viditelnosti uživatelů sedících v dálce. V jiných případech není potřeba zvětšovat rozlišení snímků, jako například aplikace používající postavu pro komunikaci s uživatelem sedícího těsně před obrazovkou.

3.2.4 Časová délka snímku

V aplikaci „Microsoft Agent Character Editor“ lze časové kvantum jednotlivým snímkům přiřazovat libovolně od 1 setiny sekundy až do hodnoty 665 setin sekundy. Toto nastavení se u jednotlivých snímků může lišit v závislosti na provedené akci. U větších pohybů celým tělem jsem používal dobu trvání jednoho snímku okolo 5 milisekund, což je rovno 20 snímků za vteřinu. Tato frekvence již zajistí kvalitní plynulý pohyb. Ale jsou případy, kdy postava pouze mrkne. Zde postačí doba trvání jednotlivých snímků trvajících okolo 80 setin sekundy.

Editor postav podporuje nastavení větvení animace u konkrétního snímku. U každého snímku lze definovat až tři různé větve s možností nastavení pravděpodobnosti. Větvení umožňuje vytvářet animace, které se mění během jejich spuštění a vytváří i smyčky. Toto může způsobovat problémy při přehrání jedné animace za druhou. Například při přehrávání animace looping nebo větvení animace by mohla pokračovat do nekonečna, pokud se nepoužije metoda „Stop“.

Snímky neobsahující žádné obrazy a jejich doba trvání je nastavena na 0 se nezobrazí. Tato funkce se dá využít k vytvoření snímku, který podporuje větvení bez toho, aby byl snímek viditelný. Je tedy zapotřebí nastavit doby trvání na 0.

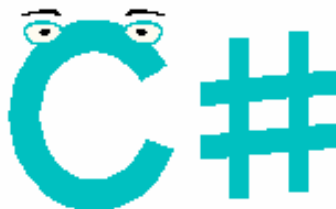
3.2.5 Snímkový přechod

Při navrhování animace je také potřeba zvážit přechod mezi jednotlivými snímky. Například při vytváření animace postavy gestikulující napravo a jiné gestikulující nalevo, je potřeba animovat postavu hladce z jedné pozice do druhé. Přestože by se dalo vytvořit toto v jediné animaci, nabízí se lepší řešení a to rozdělení na dvě animace a používat neutrální polohu do které se postava vrací. Tato neutrální poloha by tak měla být zahrnuta v každé animaci, do které se pokaždé vrátí. V „Microsoft Agent Charakter Editor“ se může zadat doplňující „Return“ animace pro každou animaci postavy. Návratová animace obvykle nepřesahuje 4 snímky. Z toho vyplývá že návratová animace do neutrální polohy je rychlá.

Například s použitím „gestikulovat vpravo“ pak „gestikulovat vlevo“ se může vytvořit „GestureRight“ počínaje snímkem, kde postava je v neutrální poloze a pak stačí přidat snímky rozšiřující se pravé ruky na pravé straně. Dále vytvořit pro tuto animaci návratovou animaci s použitím snímků předchozí animace, která vrátí postavu zpět do neutrální polohy. Lze toto přiřadit jako návratovou animaci ke „GestureRight“. Následně je potřeba vytvořit animaci „GestureLeft“, začínající opět od neutrální polohy a stejným způsobem jako u předchozí animace. Návratová animace obvykle začíná obrazem, který je posledním snímkem předchozí animace.

Spuštění a návrat do stejné neutrální polohy, ať už v rámci animace nebo pomocí Return animace, umožňuje přehrát různé animace v libovolném pořadí. Microsoft Agent animační služby automaticky přehrávají určené návratové animace v mnoha situacích.

Například animační služby spustí určené návratové animace před spuštěním animace postavy v klidovém stavu („Idling“). Návratové animace je tedy vhodné používat pokud animace již nebudou končit v poloze neutrální.










Obrázek 4 - Neutrální poloha Microsoft Agent použitého v bakalářské práci

3.2.6 Animace pohybu úst

Aplikace „Microsoft Agent Character Editor“ rovněž obsahuje sadu obrázků úst pro animování mluvení postavy. Obecně platí, že pohyb úst je velmi důležitý. Postava se může jevit méně sympatická nebo věrohodná, pokud se ústa pohybují asynchronně s hlasovým výstupem. Obrázky úst v aplikaci umožňují postavě přesnou synchronizaci otevírání úst k hlasovému výstupu. Vzorové obrázky se mohou předefinovat vlastními a celý tento mechanismus funguje tak, že postava čte text z textové bubliny a podle písmen a slabik se ústa otevírají. Při použití „text-to-speech engine“ je výsledkem hlasový výstup. Jediná nevýhoda je, že tento pohyb úst je převážně podporován pro anglicky mluvený výstup.

Tabulka 1 - Polohy úst v aplikaci „Microsoft Agent Character Editor“

Pozice pusy	Vzorový obrázek	reprezentace
Zavřená		Normální tvar zavřených úst.
Otevřená 1		Ústa jsou pootevřena na celé šířce.
Otevřená 2		Ústa jsou částečně otevřena na celé šířce
Otevřená 3		Ústa jsou otevřena, v plné šířce.
Otevřená 4		Ústa jsou zcela otevřena na celé šířce.
Otevřená středně		Ústa jsou otevřena na polovinu šířky.
Otevřená úzce		Ústa jsou otevřena na malou šířku..

Zdroj: [4]

3.3 Stav Agenty

„Microsoft Agent“ animační služby automaticky přehrávají některé animace. Například při použití příkazu „moveTo“ nebo „GestureAt“ animační služby přehrají odpovídající animaci. Podobně při nečinné době služby automaticky spustí určenou animaci pro nečinnost agenta. Na podporu těchto stavů lze definovat odpovídající animace a pak přiřadit ke stavům. Tyto animace lze opakovaně přehrávat libovolně použitím funkce „Play“ i když jsou přiřazeny do určitého stavu.

Lze přiřadit i více animací do stejného stavu a animační služby budou náhodně vybírat jednu z animací. Toto umožňuje postavě více přirozené a rozmanité chování.

Ačkoliv je možné přiřadit stavům animace s větvenými snímky, je dobré se vyhnout „looping“ animacím (smyčkové animace). V opačném případě je nutné použít metodu „Stop“ než bude moci být spuštěna další animace.

Je důležité definovat a vytvořit alespoň jednu animaci pro každý stav co nastává pro postavu. Pokud se nenadefinují k těmto stavům animace, tak postava se nebude chovat přiměřeně k uživateli. Nicméně, jestliže stav se nevyskytuje při používání dané postavy tak se pochopitelně nemusí přiřazovat animace. Například v případě aplikace u níž se nevolá metoda „moveTo“, není třeba tvořit animaci a přiřazovat k tomuto stavu.

Tabulka 2 - Stav postavy „Microsoft Agent 2.0“

Stav	Příklad užití
GesturingDown	Když je zpracována animační metoda GestureAt .
GesturingLeft	Když je zpracována animační metoda GestureAt .
GesturingRight	Když je zpracována animační metoda GestureAt .
GesturingUp	Když je zpracována animační metoda GestureAt .
Hearing	Když je detekován začátek mluveného vstupu.
Hiding	Když uživatel nebo aplikace skryje postavu.
IdlingLevel1	Když postava začne prostoje.
IdlingLevel2	Když postava začne prostoje druhé úrovně.
IdlingLevel3	Když postava začne prostoje třetí úrovně.
Listening	Když postava odstartuje poslouchání (uživatel stiskne klávesu pro hlasový vstup).
MovingDown	Když je zpracována animační metoda MoveTo .

MovingLeft	Když je zpracována animační metoda MoveTo .
MovingRight	Když je zpracována animační metoda MoveTo .
MovingUp	Když je zpracována animační metoda MoveTo .
Showing	Když uživatel nebo aplikace zobrazí postavu.
Speaking	Když je zpracována animační metoda Speak .

3.3.1 Stav „Hearing“ a „Listening“

Přiřazená animace k Listening stavu se přehrává, když uživatel stiskne horkou klávesu pro hlasový vstup. Při této animaci by měla postava vypadat zaujatě a pozorně. Animace pro stavy „Hearing“ a „Listening“ mohou být obdobné. Animace nastavená na stav „Hearing“ se spustí při hlasovém vstupu uživatele. Návrh této animace by měl být jednoduše navržený proto, aby uživatel věděl, že postava mu aktivně naslouchá. Různá gesta a přitakání postavy jsou vhodná. Pro zajištění přirozené variability je dobré navrhnout více animací pro tento stav postavy.

3.3.2 Stav „Gesturing“

Pokud se plánují u postavy ukazující gesta, je třeba tyto gesta neanimovat a přiřadit je k daným stavům. Animace přiřazené ke gestikulujícím stavům se přehrávají při volání metody „GestureAt“. Pokud jsou pro postavu nadefinována ústa, je možnost i pohybu úst při těchto animacích.

Animační služby určují umístění postavy a její vztah k souřadnicím specifikované v metodách a přehrávání vhodných animací. Gestikulující směr je vždy s ohledem na charakter. Například jestliže je zavolána animace „GestureRigth“, tak postava by správně měla ukazovat pravou rukou doprava.

3.3.3 Stav „Showing“ a „Hiding“

Stavy „Showing“ a „Hiding“ spouští přidělené animace, když uživatel nebo žádosti hostitelské aplikace zobrazí nebo skryjí postavu. Tyto stavy je důležité použít při viditelném režimu postavy. Při definování animací pro tyto stavy, je důležité mít na paměti, že postavy se mohou objevit nebo zmizet kdekoliv na obrazovce. Protože uživatel má možnost zobrazit nebo skrýt libovolnou postavu, je nutné tyto stavy vždy definovat alespoň jednou animací.

Animace přiřazené stavu „Showing“ obvykle končí snímkem neutrální polohy postavy. Naopak animace stavu „Hiding“ začíná v neutrální poloze postavy. Animace stavů „Showing“ a „Hiding“ může obsahovat prázdný snímek na začátku a na konci. Přirozeně u „Hiding“ stavu prázdný snímek na konci animace a v druhém případě na začátku animace.

3.3.4 Stav „Idling“

Stavy „Idling“ jsou progresivní. Animační služby začínají používat animace pro „Level1“ po krátké době animace a po delší době nastává spouštění animací pro „Level2“. Po této nečinnosti cyklus postupuje do „Level3“ a v tomto stavu zůstane až do zrušení. Například spuštění nové animace.

Animace pro stav „Idling“ má vyjadřovat neaktivitu postavy, ale zároveň agent by neměl rozptylovat uživatele. U těchto animací je důležité odrážet reakce postavy důvtipem, však jasným způsobem. Animace jako mrkání očima a rozhlížení je vhodné přiřadit pro stav „IdlingLevel1“. Pro stav „IdlingLevel2“ lze použít animaci čtení knihy nebo zívání. Ideální animace pro „IdlingLevel3“ může být spánek či poslech hudby ze sluchátek. Všechny tyto animace pro klidový režim jsou převážně bez velikého pohybu kvůli rozptylování uživatele. Protože animace stavu „Idling“ jsou často používány, je třeba mít více variant pro tento stav i pro různé jeho stupně. A to zejména pro „IdlingLevel1“ a „IdlingLevel2“, které se často opakují.

Stavy „Idling“ jsou navrženy k vyhnutí se situaci, kdy postava nepřehrává žádnou animaci. Obrázek postavy, která se nemění po krátké době, ubírá věrohodnosti a interaktivity. Přitom zachování věrohodnosti není tak náročné. Někdy stačí mrknutí, viditelný dech nebo posun těla.

3.3.5 Stav „Speaking“

Animační služby „Microsoft Agent“ používají stav „Speaking“ ve chvíli použití „text-to-speech engine“. Dalším způsobem je animování pohybu úst postavy a přiřadit je do tohoto stavu. Animace je spuštěna ve chvíli čtení textu s balonového dialogu.

3.3.6 Stav „Moving“

Stav „Moving“ se používá v aplikaci při zavolání metody „MoveTo“. Animační služby samy určí animaci k přehrání podle současné polohy a cílové polohy uvedené například souřadnicemi v pixelech. V mé bakalářské práci používá moje postava pro všechny čtyři směry stejnou animaci. Z toho vyplývá že v tomto případě nezáleží na poloze. Jinak je důležité se při návrhu směrů opět řídit směry z pohledu postavy. Pokud aplikace nepoužívá metodu „MoveTo“ tak není třeba animovat posun postavy.

Animace stavu „Moving“ by měla oživit postavu při posuvu na obrazovce. Poslední snímek této animace je zobrazován během doby přesunu přes obrazovku. Během této doby se postava nemění a až v cílovém bodě se použije návratová animace do neutrální polohy.

3.3.7 Standardní animační sada

I když lze vytvořit postavu s vlastními názvy animací, „Microsoft Agent“ definuje vlastní standardní názvy animací. Postavy s těmito názvy mohou být použity jako výchozí postava. V následující tabulce jsou uvedené pouze některé důležitější animace. Celkem jich je 59.

Tabulka 3 - Standardní animace „Microsoft Agent 2.0“

animace	příklad užití	příklad animace.
Blink	Při ukončení mluvení nebo při nečinnosti	Mrknutí očima.
GestureDown 1,2	Když postava potřebuje ukázat dolů.	Gestikulace rukou dolů.
GestureLeft 1,2	Když postava potřebuje ukázat doleva.	Gestikulace doleva.
GestureRight 1,2	Když postava potřebuje ukázat doprava.	Gestikulace doprava.
GestureUp 1,2	Když postava potřebuje ukázat nahoru.	Gestikulace rukou nahoru.
Hearing 1,2	Když postava slyší mluvený projev.	Přiklonění hlavou.
Hide	Při vypnutí Microsoft Agentu.	Vystoupení dveřmi.
Idle1_1	První level nečinnosti.	Mrknutí
Idle2_1	Druhý level nečinnosti.	Zívání nebo čtení.
Idle3_1	Třetí level nečinnosti.	Spánek
LookDown	Když postava potřebuje pohled dolů.	Natočení očí dolů.
MoveLeft	Když postava potřebuje pohyb doleva.	Pootočení postavy a skok.
Show	Při zapnutí Microsoft Agentu.	Příchod dveřmi

Tabulka 4 - Animace přiřazené ke stavům postavy

Stav	Požadovaná animace
GesturingDown	GestureDown
GesturingLeft	GestureLeft
GesturingRight	GestureRight
GesturingUp	GestureUp
Hearing	Hearing1, Hearing2
Hiding	Hide
IdlingLevel1	Blink, Idle1_1, Idle1_2
IdlingLevel2	Blink, Idle1_1, Idle1_2, Idle2_1, Idle2_2
IdlingLevel3	Idle3_1, Idle3_2
Listening	Alert
MovingDown	MoveDown
MovingLeft	MoveLeft
MovingRight	MoveRight
MovingUp	MoveUp
Showing	Show
Speaking	RestPose

3.4 Zásady animace

Efektivní návrh animace vyžaduje víc než jen vykreslování postav. Úspěšní animátoři sledují celou řadu zásad a technik pro vytvoření „věrohodné“ postavy.

3.4.1 Technika „Squash and Stretch“

Jak již z názvu techniky vyplývá, jde o deformaci animovaných objektů. Poskakující míč po podlaze se zdánlivě stlačuje a při odrazu natahuje. Stupeň stlačení a roztáhnutí míče vypovídá o jeho tvrdosti. Při použití této techniky lze snadno sdělit divákům o základních vlastnostech tělesa. Tedy těleso vypadá mnohem více realisticky.

Začátek „Squash and Stretch“ byl již ve 20. letech 20.století. Ukázky jsou dnes ve většině animovaných grotesek jako je „Tom and Jerry“ a další známé animované filmy.

3.4.2 Očekávání

Očekávání vytváří půdu pro novou akci. Bez předběžných akcí tělesné pohyby vypadají prudké, tuhé a nepřírozené. Tento princip je založen na tom, jak se tělo pohybuje v reálném světě. Pohyb v jednom směru často začíná s pohybem ve směru opačném. Nohy se před skokem přikrčí nebo před vydechnutím je třeba se nejprve nadechnout. Očekávání nepomáhá pouze připravit publikum na akci, ale také projev povahy postavy. Klíčovým aspektem vytvořené věrohodné postavy je demonstrování plánované akce postavy. Tím je myšlena přirozeně se rozvíjející akce s promyšleným záměrem.[3]

3.4.3 Načasování

Načasování definuje povahu akce. Rychlost pohybu hlavy zleva doprava vyjadřuje zda postava se rozhlíží okolo nebo dává najevo negativní reakci. Dalo by se říct, že načasování také zprostředkovává hmotnost a velikost objektu. Větší objekty mají tendenci trvale déle zrychlovat a zpomalovat než menší objekty. Navíc typ chůze postavy ovlivní jak postava upoutá diváka nebo uživatele.[3]

3.4.4 Inscenace

Postava může využít pozadí i rekvizity pro sdělení své nálady nebo účelu. Inscenace je také to, co nosí postava, světelné efekty, zorný úhel a přítomnost dalších postav. Tyto prvky přispívají k posílení osobnosti postavy.[3]

3.4.5 „Slow In-and-Out“

Název metody se dá volně přeložit jako pomalý začátek a konec. Tato metoda se dá popsat jako pohyb postavy z jednoho místa hladce do druhého. Postava začíná a končí akce pomalu. Metoda spočívá ve změně časové frekvence snímků během jedné akce. První a poslední snímky mají delší časový trvání. Tedy pohyb zprvu má pomalý charakter a na dále zrychluje k určitému bodu, kde opět začne plynule zpomalovat.[3]

3.4.6 Vektor pohybu

Živé bytosti v přírodě zřídka kdy se pohybují v dokonale rovné přímce. Jako výsledek je oblouková nebo křivočará cesta. I tento pohyb vnese do postavy přirozenější chování. Velikost oblouku má spojitost s rychlostí, neboť rychlejší pohyb způsobí plošší zakřivení oblouku.[3]

3.4.7 Zveličování

Dobří animátoři často zveličují tvar, barvy, emoce či akci postavy. Například postava natáhne svoji paži do té míry, že se jeví pružná. Avšak přehánění musí být vyvážené. Jestliže se používá zveličování v některých situacích a v dalších ne, přehnaná

akce se může zdát nereálná a byla by od uživatele třeba i chápána se zvláštním významem.[3]

3.4.8 Sekundární akce

Animace vyžaduje více než mechanické vytváření spojení obrázků z jedné pózy do druhé. Primární akce jsou obvykle podporovány akcemi sekundárními. Sekundární akce napomáhají zlepšit prezentaci, ale neměla by odvádět pozornost od primární akce. Výrazy tváře lze dobře využívat jako druhotné akce pohybu těla. Bohatství přichází z přidávaných prvků, které podporují hlavní myšlenku.

3.4.9 Solidní kresba

Vytvoření animované postavy zahrnuje více než série snímků. Efektivní návrh animace se zabývá i jak postava vypadá v různých pozicích a z různých úhlů. Postavy poskytované ve dvourozměrných obrázcích se stávají mnohem realističtější a věrohodné pokud jsou koncepčně ve třech rozměrech. Pokud možno je dobré se vyhnout zrcadlení obličeje, paží a nohou na obou stranách těla. Následkem jsou tuhé a nepřirozené prezentace.

3.4.10 Apelování

Úspěšná realizace závisí na dobrém pochopení publika. Celková „image“ postavy by měla apelovat na cílovou skupinu. Apelování nevyžaduje „foto-realismus“. Osobnost postavy může být zprostředkována bez ohledu na to jaký má jednoduchý tvar pomocí gest, držení těla a dalšími pohyby. Při návrhu postavy je jako první třeba navrhnout různorodé výrazy, ať třeba pro pytel cukru, ručník nebo malý kobereček. Postavy jednoduchých tvarů jsou často účinnější než ty složité. Mnoho známých postav disponuje pouze třemi prsty a jedním palcem.

3.5 Uživatelské rozhraní

Hlavní záměr při vývoji „Microsoft Agent“ bylo vytvořit pomocníka, který interaktivně bude pomáhat uživatelům. A to tak, že bude schopen rozeznat hlasový vstup uživatelů a opět odpovědět pomocí hlasového výstupu. Nyní Microsoft od této technologie opouští a v nové verzi Windows 7 již není možno spouštět MS Agenty, tedy alespoň bezchybně. „Microsoft Agent“ používá i další prostředky ke komunikaci s uživateli uvedeny níže pod odstavcem.

3.5.1 Ikona v panelu nástrojů

Má-li postava od svého autora vytvořenou ikonu, ikona bude zobrazena v oznamovací oblasti hlavního panelu, když je postava spuštěna. Tato ikona poskytuje přístup k „pop-up menu“ postavy umožňující přístup ke globálním příkazům jako jsou „Hide“ nebo „Show“.



Obrázek 5 - Ukázka ikony Microsoft Agent „Merlina“ na hlavním panelu systému Windows

Zdroj: [4]

Ukazatel myši pohybující nad ikonou v hlavním panelu zobrazí tip okno s jménem postavy. Jestliže je postava skryta, postačí jedenkrát kliknout na ikonu pro zobrazení postavy. Operace spojené s dvojitým kliknutím na ikonu závisí na aktuální aplikaci ovládající postavu.

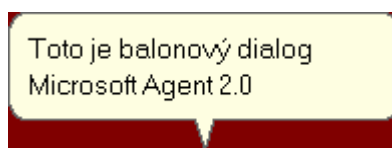
Pravým kliknutím na ikonu se zobrazí pop-up menu. Když je postava vidět, pop-up menu zobrazí stejné příkazy jako v případě kliknutím pravým tlačítkem myši přímo na postavu. V případě skryté postavy pop-up menu poskytne pouze Open(nebo Close) a otevření okna s hlasovými příkazy (Pokud jsou k dispozici).

3.5.2 Textové balóny

Kromě mluveného audio výstupu, „Microsoft Agent“ rozhraní podporuje textový výstup ve formě textových balónů používající styl z kreslených komiksů. Text uvnitř balónu je používán zároveň pro hlasový výstup. Po přečtení textu je balón skryt.

Styl balónu lze nastavit v aplikaci „Microsoft Agent Character Editor“ a to font písma, velikost písma, barvu písma, barvu pole a barvu rámečku.

Pokud je potřeba použití balónu formou dialogového okna obsahující tlačítko ok či storno a další nástroje pro řízení běhu programu, lze pro agenta použít třídu „Assistant“ s „Microsoft Office“. Ovšem v mém zadání nebylo možné použití z hlediska odlišných vlastností tohoto pomocníka importovaného přímo s „Microsoft Office“. Tento agent se nezobrazuje na vrchu obrazovky, nýbrž na vrstvě spuštěné aplikace. Má podobné vlastnosti jako standardní dialogové okno. Využití se najde u jakékoliv oknové aplikace u níž není potřeba spouštět další okna, jako v mém případě okno aplikace „PowerPoint“.



Obrázek 6 - Ukázka balónového dialogu „Microsoft Agent 2.0“

3.5.3 Software pro rozpoznání hlasu

Lze na některých stránkách stáhnout kompatibilní software, který umí rozeznat příkazy z hlasového vstupu. Seznam těchto příkazů je zobrazen v Hlasovém příkazovém okně.

4 Postup animace v praktické části

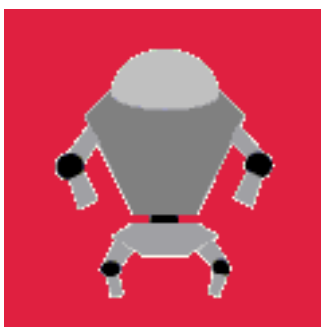
4.1 Tvorba snímků

Pro tvorbu snímků jsem vybral „Adobe Photoshop CE 7.0“ a pro další drobnou úpravu jsem používal program „Malování“ ze systému Windows, ale lze prakticky použít jakýkoliv grafický nástroj.

Důležitý aspekt pro výběr byl určitě nástroj „pravítko“ přilehlé podél stran pracovního okna, které lze nastavit na libovolné jednotky. V mém případě šlo o pixely. Další nástroj „vodící čáry“ pomocí nichž jsem byl schopen regulovat velikost, postavení a pohyb postavy na přesnost 1 pixelu. Díky tomuto lze stále kontrolovat pozici postavy v prostoru i její velikost.

Výhoda v „Adobe Photoshop“ spočívá i ve zpracování více vrstev, kde lze ukládat do vrstev různé typy očí, úst či prstů. Tyto vrstvy se snadno skryjí a z vrstev zobrazujeme jen ty potřebné.

Moje první práce započala pomocí nástroje „vrstva tvaru“. Jedná se o nástroj využívající vektorové grafiky. S pomocí nástroje „přímý výběr“ lze vektorovými tvary libovolně pohybovat. U mého prvního konceptu byla hned použita tzv. „kostra“. Tato kostra se skládala z mnoha vektorových tvarů jako jsou čtyřúhelníky, trojúhelníky, elipsy a další tvary. Kostra vlastně byla silueta postavy s pohyblivými částmi těla jako ruce, nohy, trup a hlava. V jiných vrstvách se nacházely oči, ústa a protože se jednalo o postavu robota, tak i různé knoflíky a diody. Výsledkem byla postava, se kterou lze libovolně pohybovat pomocí nástroje „přímý výběr“ a nadále snímek po snímku ukládat do 16 bitového formátu BMP.



Obrázek 7 - Ukázka negativního vlivu antialiasingu na animovanou postavu

Ovšem při spuštění animace na červeném pozadí byl podhalen špatný vliv metody vyhlazování způsobující okolo postavy obrys ve světle šedých odstínech. Nástroj „vrstva tvaru“ nepodporuje vypínání metody vyhlazování, proto jsem tuto metodu vypustil pro zdlouhavé retušování světle šedých až téměř bílých obrysů okolo postavy.

Vhodná funkce „Adobe Photoshopu“ u níž je možnost vypnutí „antialiasingu“ se nazývá „cesty“. Jedná se také o práci s vektorovou grafikou. Dokonce i nástroje zůstávají stejné. Jediný rozdíl spočívá v ukládání. U nástroje „vrstva tvaru“ probíhalo ukládání každého tvaru do jednotlivé vrstvy, zatím co u cest lze libovolně ukládat vše do jediné vektorové plochy nebo do více ploch. Rozdíl je i ve vykreslování. U cest lze vykreslit až po volbě nástroje tlačítkem „vytažení cesty“. U „vrstva tvaru“ docházelo vytažení a zároveň k vyplnění ihned po změně pozice jediného tvaru.

S novou technologií postupu jsem změnil i postavu. Zvolil jsem cestu jednoduššího tvaru, inspirovaný agentem „Svorka“. Tento agent je relativně jednoduchý, ale zároveň efektivní a oblíbený. Můj nový agent vhodný svým tvarem pro reprezentaci naší fakulty má tvar písmena „C“ a znaku „sharp“. U toho tvaru lze použít jednoduché kroucení a deformace postavy za účelem gestikulace a komunikace s uživatelem..

Pro tuto postavu stačilo vytvořit pouze dvě cesty. V první cestě byl již zmíněný tvar písmene „C“ se znakem „sharp“. Druhá cesta obsahovala pouze vertikální dlouhý obdélník znázorňující blikající kurzor na psaní písmen. Vytažení cesty s nástrojem tužka o průměru 1 pixel došlo k vytažení obrysu požadovanou barvou bez známky antialiasingu. Po každém vytažení cesty stačilo vymazat nástrojem guma vzniklé rastrové obrazce a znovu pohnout cestou do libovolné polohy či tvaru.

Vrstev bylo hned několik. Všechny ovšem obsahovaly oči ve všech možných pozicích. Na začátku zabraly více práce, ale na konec se ušetřilo hodně času. Před uložením obrázku do 16 bitové bitové mapy vždy zůstaly na pracovní ploše vybrané oči spolu s vztaženou cestou. Na obrázku 6 je ukázána ukázka polotovaru agenta „CSharp“ při gestikulaci vpravo.



Obrázek 8 - Ukázka polotovaru animované postavy „Sharp“

Tyto polotovary uloženy v 16 bitové bitmapě byly otevřeny v programu malování a přeloženy do 8 bitového BMP formátu. Následující krok spočíval v rozlité barvy do bílých ploch mezi světle modrými vytaženými cestami. Tento postup se opět provedl pro každý snímek animace. Pro tento úkol malování naprosto vyhovovalo díky jednoduchosti programu a rychlému použití. Navíc „Adobe Photoshop“ nepodporuje přímé ukládání do 8 bitového formátu BMP.

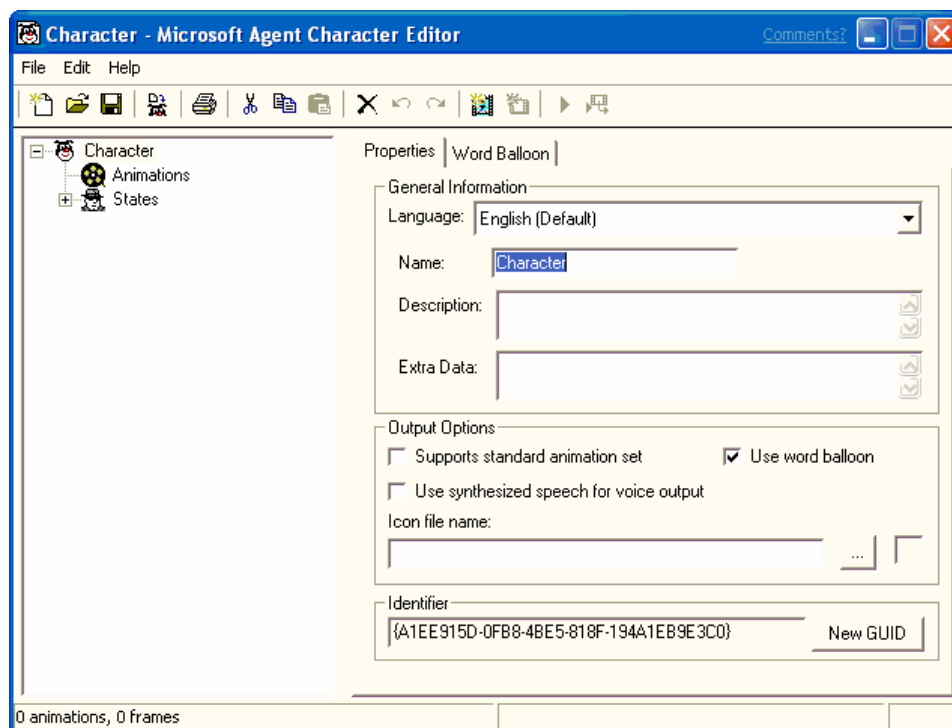
4.2 Řazení snímků pomocí aplikace Microsoft Agent Character Editor

„Microsoft Agent Character Editor“ je aplikace umožňující sestavení animace postav pro použití „Microsoft Agent“. Lze definovat animace pomocí importu 8 bitových bitmapových obrázků, nastavení jejich trvání a případně i odbočení. Aplikace dále poskytuje pro dané animované sekvence zvukové efekty včetně mluveného slova postavy.[5]

U jednotlivých postav je možnost definovat název postavy a popis. Dále jsou možnosti výstupů „text-to-speech“, syntetizovaný hlasový výstup, „pop-up menu“ a textové balónkové okna.[5]

4.2.1 Aplikace „Microsoft Agent Character Editor“

Instalace aplikace je rychlá a jednoduchá. Navíc funguje i ve „Windows 7“. Program jde snadno spustit z nabídky Start nebo z adresáře Microsoft Agent nacházející se v adresáři „Program files“. Po otevření programu se zobrazí na levé straně stromová struktura programu. Aplikace na první pohled je jednoduchá a většina funkcí lze vydedukovat bez příručky. Ovšem na některé funkce je potřeba pročíst manuálové stránky nacházející se na webových stránkách.[5]



Obrázek 9 - Ukázka aplikace „Microsoft Agent Character Editor“

Zdroj: [5]

Kořen stromové struktury je položka „Charakter. Ta se dělí na Animations (animace) a States (stavy). Složka Animace se dále dělí na dílčí animační sekvence a ty dále na jednotlivé snímky. Ke každé animační sekvenci lze přiřadit libovolný počet stavů

nacházející se ve složce „States“ i jednotlivé zvukové formáty typu „WAW“. Produktem této aplikace jsou dva formáty souborů. Ten první je s příponou „ACD“ sloužící k editaci postavy a je kdykoliv možnost zavření a otevření k další editaci. Druhý formát s příponou „ACS“ je hotový vygenerovaný „Microsoft Agent“ připraven k použití, avšak tento soubor už nelze otevírat a editovat. [5]

Aplikace podporuje dva formáty pro import obrázků. Již na začátku kapitoly jsem se zmínil o podpoře formátu 8 bitového BMP. Dalším podporovaným formátem krom „BMP“ je „PNG“. „PNG“ používá bezeztrátovou kompresi rastrové grafiky. Byl vyvinut jako zdokonalená verze GIFu. Na rozdíl od GIFu lze použít až 24 bitovou barevnou hloubku, avšak co je důležité, „Microsoft Agent Character Editor“ a tedy i „Microsoft Agent“ podporuje právě pouze 8 bitovou barevnou hloubku. V případě, že postava používá jen několik odstínů, je její komprese velká. Praktická nevýhoda použití tohoto formátu spočívá v tom, že všechny obrázky použité pro postavu musí mít naprosto stejnou paletu barev. Jakmile se barevná paleta liší jen o jeden odstín, dojde k protečení tohoto odstínu na odstíny ostatní. Například mě se stalo, že animace byla černobílá.

Po tomto nezdaru jsem přešel na formát BMP, u kterého se paleta nemění a zůstává pro všechny snímky stejná. Snímky zabírají mnohem více místa, ale dnešní možnosti PC a sítě internet jsou schopny unést větší nároky. Vygenerovaný soubor „ACS“ se zhruba 12-ti animacemi zabírá okolo 250 KB, což je v dnešní době velice málo.

4.3 Animační sekvence

Moje postava „Csharp“ obsahuje pouze několik základních animačních sekvencí, aby byly pokud možno animovány všechny stavy (States) Microsoft Agent. Níže je uveden přehled animací.

4.3.1 „Start“

Tato animace je použita při zobrazení Microsoft Agent a přiřazena ke stavu „Showing“. Zobrazení agenta probíhá jako psaní znaků na obrazovce v nějakém textovém editoru. Animace probíhá napsáním pomocí blikajícího kurzoru písmena „C“ se znakem „#“. Po dopsání se agent nachází již ve výše uvedené neutrální poloze.

4.3.2 „Konec“

Animace použitá při skrytí agenta z obrazovky a přiřazena ke stavu „Hiding“. Stejně jako u zobrazení je použit blikající kurzor textového editoru, ale tentokrát s použitím klávesy „Backspace“, která se chvíli přetlačuje s agentem a po té ho smaže z řádku. Obě tyto animace jsou jednoduché, nenáročné na čas, avšak vypadají efektivně.

4.3.3 „Mrknutí“

Velmi jednoduchá animace přiřazená ke stavu „IdlingLevel1“. Ve chvíli krátké neaktivity Microsoft Agent dojde občas k mrknutí jeho očí. Tato animační sekvence se

skládá ze tří snímků určených k přirozenému chování agenta, stejně jako následující animace.

4.3.4 „Neaktivni1“

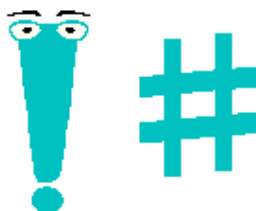
Druhá animace přiřazená ke stavu „IdlingLevel1“. Obě animace přiřazené k tomuto stavu jsou náhodně střídány pro zachování přirozenosti a věrohodnosti Microsoft Agent. Pro tento stav je nejlepší použít pokud možno co nejvíce krátkých animací pro věrohodnost agenta. Animace probíhá tak, že agent zpozorní a rozhlíží se očima ze strany na stranu.

4.3.5 „Neaktivni2“

Animace je přiřazená ke stavu „IdlingLevel2“ a spustí se po delší době neaktivity agenta. V mém případě animace probíhá tak, že Microsoft agentovi se pomalu přivírají oční víčka a následuje spánek s velikými nádechy a výdechy. Opět i zde platí pro tento stav co nejvíce animací. Moje postava je určená k prezentaci „PowerPoint“ a tedy ani není tento stav nutný, nicméně kvůli větší univerzálnosti není na škodu tuto animaci vytvořit.

4.3.6 „Vystraha“

Animace není přiřazena k žádnému stavu. Je navržena pro případy zdůraznění důležité informace na obrazovce nebo obecně pro varování uživatele. V této animaci se transformuje písmeno „C“ postavy na vykřičník se zvukovým efektem znějící jako cinknutí s vysokým tónem.



Obrázek 10 - Ukázka s animační sekvence „Vystraha“

4.3.7 „Naslouchani“

Tato animace je použita u stavu „Listening“. Tedy agent provede animační sekvenci v případě detekování hlasového vstupu. Samozřejmě tuto animaci lze použít i na další zdůraznění a gestikulaci. Animace vypadá tak, že písmeno „C“ trochu přikloní svoji pravou stranu, ze které vyroste ucho, mezitím znak „#“ se promění ve veliký otazník.

4.3.8 „Mluvení“

Tato animace je navržena pouze pro dubbovaný výstup a není přiřazena k žádnému stavu. U této animační sekvence dochází pouze k pohybu úst agenta.

4.3.9 „Pohyb“

Animace navržena pro pohyb napříč obrazovkou. Přiřazena je ke stavu „MovingUp“, „MovingLeft“, „MovingRight“ a „MovingDown“. Animace probíhá tak, že se nejprve scvrkne znak „#“ do malé kostky pohybující se do středu písmene „C“, které se následně také scvrkne až úplně zmizí. Pohyb agenta po obrazovce probíhá skrytě.

4.3.10 „KonecPohybu“

Animace „KonecPohybu“ je animací návratovou k animaci Pohyb. Probíhá úplně stejně, pouze v opačném pořadí.

4.3.11 „UkazaniNahoru“

Animace přiřazená ke stavu „GesturingUp“. Tato animace patří stejně jako další gestikulace k náročnějším animacím, skládající se z více než 36 snímků. Průběh animace začíná v okamžiku deformace znaku „#“ na šipku otáčející se směrem vzhůru, která se zastaví ve vertikální poloze a dvakrát se posune nahoru a dolů pro zvýraznění směru. Návrat animace probíhá stejně jako v první části animace, jen mnohem rychleji s použitím menšího počtu snímků. Jako doprovodnou animaci ke gestikulaci šipkou vzhůru postava používá pohled očima vzhůru ve směru gestikulace.

4.3.12 „UkazaniDoprava“

Animace přiřazená ke stavu „GesturingRight“. Další z náročnějších animací obsahující 29 snímků. Animace opět používá deformaci znaku „#“. Tentokrát v hranatý projektil letící z pravé strany obrazovky do levé skrz agenta. Na jeho pravé straně po průletu se změní v šipku ukazující doprava z jeho pohledu. Zpětná animace používá stejné snímky. I zde bylo použito pohybu a mrknutí očí při průletu skrze písmeno „C“.

4.3.13 „UkazaniDoleva“

Animace přiřazená ke stavu „Gesturing Left“. Z 36 snímků skládající se animace je založena na stejném principu deformace postavy jako dvě předchozí. Zdeformovaný znak „#“ v šipku se otočí hrotem doleva s agentova pohledu a po zamrknání očima a jejich následným pohledem vlevo šipka dvakrát zagestikuluje stejným směrem. Návratová animace založena na stejných snímcích proběhne mnohem rychleji.

4.3.14 „UkazaniDolu“

Tato animace přiřazená ke stavu „GesturingDown“ se od předchozích gestikulací liší. Zde není použit znak „#“, ale místo něho se deformuje spodek písmene „C“. Spodní

část je transformována na šipku ohýbající se směrem dolů. Ocásek postavy spolu s očima ukáží směrem dolů a po nedlouhé době dochází ke zpětné animaci. Animace obsahuje 38 snímků.

Tabulka 5 - Tabulka animací postavy „Csharp“

Animace	Návratová animace	Zvukový efekt	Přiřazení ke stavu
Start	ne	Ne	Showing
Konec	Ne	Ne	Hiding
Mrknuti	Ne	Ne	IdlingLevel1
Neaktivni1	Ne	Ne	IdlingLevel1
Neaktivni2	Ne	Ne	IdlingLevel2
Vystraha	Ne	Ano	ne
Naslouchani	Ne	Ne	Ne
Mluveni	Ne	Ne	Ne
Pohyb	KonecPohybu	Ne	MovingLeft, MovingRight, MovingUp, MovingDown
UkazaniNahoru	Ne	Ne	GesturingUp
UkazaniDoprava	Ne	Ne	GesturingRight
UkazaniDoleva	Ne	Ne	Gesturing Left
UkazaniDolu	Ne	Ne	GesturingDown

5 Použití postavy v aplikaci

5.1 Metody knihovny „Microsoft Agent Control 2.0“

Jakmile je postava načtena, lze používat širokou škálu metod knihovny „Microsoft Agent Control 2.0“. Typicky první použitá metoda je „Show“. Ve chvíli kdy je postava načtena, je už na obrazovce, ale pouze v neviditelném režimu. Při zavolání metody „Show“ se stav postavy změní na viditelný. Zároveň se spustí animace přidělena tomuto stavu obsahující první snímek prázdný a další zobrazují pozvolný příchod postavy na obrazovku. Po celou dobu zobrazení postavy platí, že poloha postavy na obrazovce leží přes všechna otevřená okna všech aplikací, tedy postava se nachází vždy na vrcholu pracovní plochy.

Nyní po zviditelnění postavy nám umožňuje „Microsoft Agent Control 2.0“ použití metody „Play“. Parametr metody je název animace napsaný v uvozovkách. Spuštění metody má za následek přehrání dané animační sekvence.

Zavolaná metoda „Speak“ vypisuje text do textového balónu a pokud postava obsahuje program pro převod textu na hlasový výstup, tak z tohoto balónu předčítá text.

Použitím metody „MoveTo“ se postava přesune na nové místo. Při zavolání metody dojde nejprve k přehrání animace „Pohyb“ a při dokončení pohybu na dané souřadnice, které jsou parametrem funkce, je spuštěna návratová animace „KonecPohybu“.

Zavolaná metoda „Hide“ spustí animaci pro schování postavy a po té nastaví postavě neviditelný režim. Metoda funguje obdobně jako metoda „Play“, ale v opačném pořadí.

„Microsoft Agent“ zpracovává všechny zavolané animace a požadavky asynchronně. To umožňuje aplikaci vykonávat další operace, zatímco je vyřizován požadavek. Vypadá to asi tak, že program přečte všechny řádky, kde se volají dané animace a nečeká na jejich dokončení, ale pokračuje dále v běhu programu a obsluhuje jiné funkce. Spuštěné animace jsou zařazeny do fronty zpracování. Zde se jedna po druhé zpracovávají až do doby vyprázdnění fronty. Například je-li spuštěna aplikace s deseti příkazy pro animaci a jedenáctý příkaz slouží pro zobrazení „messageBoxu“, „messageBox“ bude zobrazen hned po spuštění aplikace.

Lze synchronizovat frontu animací postavy vytvořením objektu s odkazem na animační žádost. Pomocí monitorování animačních žádostí lze čekat než se provede daná animace a teprve po provedené animace začne načítat do fronty další animace. Aplikace se pozastaví u „messageBoxu“ nebo spustí animaci druhého agenta. Tato metoda se mi nejvíce uplatnila při synchronizaci dvou spuštěných agentů.

„Microsoft Agent“ přehrává některé animace automaticky. Například, pokud postava nějaký čas nemá ve frontě žádné animace, po několika sekundách začne agent přehrávat animace stavu „IdlingLevel1“, je-li povolený hlasový vstup, postava přehraje animaci pro poslech.

5.2 Implementace pomocí „Visual C#“

Prostředí Visual C# jsem vybral po konzultaci s mým vedoucím bakalářské práce, (po prostudování vlastností tohoto jazyka), a z důvodu nabídky dynamických knihoven společnosti Microsoft. Dále tento jazyk svou syntaxí je nejbližší jazyku C++, ve kterém jsem absolvoval tři semestry. S tímto programovacím jazykem jsem se setkal až při vývoji bakalářské práce.

Po dokončení mé postavy „Csharp“ můj vývoj se zaměřil na programovací část bakalářské práce. Prvním úkolem bylo zjistit jak načíst Microsoft Agentu v aplikaci.

5.2.1 Inicializace postavy

Před použitím postavy „Microsoft Agent“ jsou třeba reference na objektové knihovny „Microsoft Agent Control 2.0“. Ve vývojovém prostředí „Visual C#“ tyto reference lze získat přidáním „Microsoft Agent Control 2.0“ na „toolbox“.

Po přidáním této položky do „toolboxu“ je vytvořen automaticky objekt „axAgent1“. V této proměnné je možno uchovávat libovolný počet postav protože objekt lze použít jako pole obsahující jednotlivé postavy. Indexovat toto pole lze typem „string“, což usnadňuje práci se správou jednotlivých agentů.

Tento objekt však neovládá postavy, ale pouze je načítá. Třída, která uchovává jednotlivé postavy, se nazývá „AgentObjects.IAgentCtlCharacter“. Instance této třídy uchovává pouze jedinou postavu. Tím pádem při použití více postav je nutné vytvořit pro každou postavu jeden objekt. Níže je uveden příklad, jak taková inicializace postav vypadá v programovací jazyce „C#“. V příkladu je znázorněna inicializace dvou postav a následné jejich zobrazení pomocí metody „Show“.

```
private AxAgentObjects.AxAgent axAgent1;
private AgentObjects.IAgentCtlCharacter Csharp;
private AgentObjects.IAgentCtlCharacter Peedy;

public Form1()
{
    InitializeComponent();
    axAgent1.Characters.Load("Csharp", "C:/Csharp.acs");
    axAgent1.Characters.Load("Peedy", "C:/Peedy.acs");
    Csharp = axAgent1.Characters["Csharp"];
    Peedy = axAgent1.Characters["Peedy"];
    Csharp.Show(null);
    Peedy.Show(null);
}
```

Instance tříd uchovávací agenty je třeba pojmenovávat podle načítané postavy. Zlepší se tak přehlednost při použití více agentů. Stejně tak tyto názvy je vhodné používat pro pojmenování postavy v objektu „axAgent1“. Při volání metody „Load“ druhým parametrem je cesta k souboru Microsoft Agent. Dále v programu je již pouze používán objekt „Peedy“ a „Csharp“

Jisté potíže nastávají při používání operačního systému „Windows 7“. Společnost „Microsoft“ nadále již přestala s vývojem této technologie a další podporou. Proto u nové řady „Windows 7“ již dochází k chybám při zobrazení „Microsoft Agent“. Ovšem lze z webu stáhnout softwarovou záplatu.

5.2.2 Pohyb po obrazovce

Pohyb agenta po obrazovce je zajištěn metodou „moveTo“, jejíž první dva parametry jsou vzdálenosti v pixelech od levého horního rohu. Využití není jen v pohybu napříč obrazovkou, ale také pro určení místa polohy, kde se má postava objevit. Tuto metodu lze volat již před metodou „Show“.

Při zadávání parametru šířky a výšky někdy není vhodné užívat pixely. Vzhledem k tomu, že některé aplikace jsou ve formátu „fullscreen“, pohyb agenta by nebyl ve správném poměru při změně rozlišení obrazovky. Proto je vhodné získat rozlišení obrazovky v pixelech. Tyto pixely se už pouze vynásobí číslem od nuly do jedné. Podle toho kde chceme aby se postava vyskytovala.

```
Csharp.MoveTo(0, 0, 1);
Csharp.Show(null);
int sirka = Screen.PrimaryScreen.Bounds.Width;
int vyska = Screen.PrimaryScreen.Bounds.Height;
short width = (short)(sirka * 0.70);
short height = (short)(vyska * 0.60);
Csharp.MoveTo(width, height, 1);
Csharp.MoveTo(300, 250, 1);
```

Při volání metody „moveTo“ je automaticky volána animace pro začátek a konec pohybu. O spouštění těchto animací se zcela nemusíme starat.

5.2.3 Přehrávání animací

K přehrávání animací nám slouží metoda „Play“ s parametrem typu „string“. Každá postava má své vlastní pojmenované animace. Je důležité před použitím postavy získat seznam všech animací, které daná postava umí. Bez tohoto seznamu nelze zadávat do funkce parametr s názvem animace pokud neznáme jejich názvy. Následné animace jsou přehrávány v daném pořadí v jakém byly přečteny a uloženy do fronty animací.

```
Csharp.Play("Mrknuti");
Csharp.Play("UkazaniDolu");
Csharp.Play("Mluveni");
Csharp.Play("Mrknuti");
Csharp.Play("Vystraha");
```

5.2.4 Mluvení postavy

Postava běžně komunikuje pomocí balónových dialogů. Na internetu je možnost stáhnout „text-to-speech engine“. Ten dokáže číst text z dialogu a následně převádět na mluvené slovo. Balónový dialog vypisuje text, který je předán jako parametr funkce typu „string“. Druhý parametr, také typu „string“, se standardně nastavuje na hodnotu „null“.

```
Csharp.Speak("Ahoj, jmenuji se Csharp.", null);  
Csharp.Speak("Já jsem Microsoft Agent.", null);  
Csharp.Speak("A jak se máš ty?", null);
```

5.2.5 Synchronizace agentů

Při použití postav „Microsoft Agent 2.0“ se nevyhneme synchronizaci jejich pohybu. U postav je důležité, aby jejich pohyb byl prováděn v závislosti na právě probíhající operaci v aplikaci. Stejně tak při užití více postav nastává problém se vzájemnou synchronizací pohybu jednotlivých postav.

Nevýhoda při použití „Microsoft Agent Control 2.0“ spočívá ve způsobu čtení programu. Skriptování postavy ve zdrojovém kódu je přečteno zdánlivě najednou a uloženo do fronty animací a dále jednotlivé animace jsou přehrávány jedna po druhé. Tedy z toho plyne, že program veškeré instrukce vykonává ihned a animace časově zaostávají. Jinými slovy : jsou vykonávány se zpožděním v závislosti na časové délce všech přehrávaných animacích.

Jedním možným řešením je běh programu rozdělit na více etap zpracování instrukcí. Například použitím „messageBoxu“ nebo „switche“ v cyklu a podobně. Drobná nevýhoda použití „messageBoxu“ spočívá v tom, že se zobrazí ihned při spuštění první animace ve frontě zpracování. Jinými slovy, při zadání operace programu je „messageBox“ zobrazen zároveň se spuštěnými animacemi. Ukázka zdrojového kódu znázorňuje použití více „messageBoxu“ pro synchronizaci aplikace.

```
Csharp.MoveTo(100,50,1);  
Csharp.Show(null);  
MessageBox.Show("První část animace probíhá", "Upozornění !!!");  
Csharp.MoveTo(500,250,1);  
Csharp.Play("UkazaniDoleva");  
MessageBox.Show("Druhá část animace probíhá", "Upozornění !!!");  
Csharp.MoveTo(500,250,1);  
Csharp.Hide(1);  
MessageBox.Show("Třetí část animace probíhá", "Upozornění !!!");
```

Dalším možným řešením synchronizace je pomocí použití „switche“ uvnitř metody, která je volána například stiskem tlačítka. Ovšem lze použít jakoukoliv událost pro zavolání metody. To už závisí podle účelu aplikace. V dané metodě je „switch“ obsahující jednotlivé pořadí animací. Pro výběr větve lze použít „label“, náhodné pořadí nebo

vzestupné pořadí vzestupné. Možností je celá řada. Na uvedeném příkladě je použití stiskem tlačítka „Button1“. Při každém stisku se spouští další alternativní větev. Podobné provedení je použito v praktické části.

```
private void button1_Click(object sender, EventArgs e)
{
    Switch(cislo)
    {
        case 1:    Csharp.MoveTo(100,50,1);
                  Csharp.Show(null);
                  cislo += 1;
                  break;
        case 2:    Csharp.MoveTo(500,250,1);
                  Csharp.Play("UkazaniDoleva");
                  cislo += 1;
                  break;
        default:   Csharp.MoveTo(500,250,1);
                  Csharp.Hide(1);
                  cislo += 1;
                  break;
    }
}
```

V případě použití dvou a více agentů se setkáme s jiným problémem. Při čtení programu každá postava vlastní svoji frontu animací ke zpracování a tedy všechny postavy přehrávají animace naráz. Není tedy možno takto zajistit vzájemnou kooperaci agentů. Pro tento případ lze použít třídu `AgentObjects.IAgentCtlRequest`. Každá postava by měla mít jednu instanci této třídy. Třída slouží k rozpoznání, zda animace již byla dokončena či ne. Instance této třídy se pak předává jako parametr do funkce „Wait“.

Funkce „Wait“ volaná postavou pozastaví přehrávání animace postavy a čeká na splnění podmínky. V následující ukázce je znázorněno, jak dva Microsoft Agenti se prostřídají během přehrávání animací.

```
AgentObjects.IAgentCtlCharacter CSharp;
AgentObjects.IAgentCtlCharacter Peedy;
AgentObjects.IAgentCtlRequest CSharpReg;
AgentObjects.IAgentCtlRequest PeedyReg;
CSharp.Speak("Ahoj, já jsem Csharp.", null);
CSharpReg = CSharp.Play("UkazaniNahoru");
Peedy.Wait(CSharpReg);
Peedy.MoveTo(200, 300, 1);
Peedy.Speak("Ahoj, já jsem Peedy.", null);
PeedyReg = Peedy.Play("Acknowledge");
CSharp.Wait(PeedyReg);
CSharp.MoveTo(width,height,1);
```

5.3 Popis vývoje praktické úlohy

5.3.1 Načtení a zobrazení Microsoft Agent

V prvním kroku jsem založil projekt pomocí „Visual C#“ jako „Windows Forms Application“. Pro použití Microsoft Agent v aplikaci je třeba nejprve přidat komponentu „Microsoft Agent Kontrol 2.0“ na „Toolbox“. To lze docílit pravým kliknutím kamkoliv na „Toolbox“ a zadat v kontextovém menu položku „Choose Items“. Po zobrazení okna je potřeba přelístovat na záložku „COM Components“. Zde se nachází knihovna „Microsoft Agent Control 2.0“. Stačí před názvem knihovny zaškrtnout „CheckBox“ a potvrdit tlačítkem „OK“.

Tím byla přidána komponenta na „Toolbox“ a připravena k použití. Zároveň vznikl objekt jménem axAgent1, který je instancí třídy „AxAgentObjects.AxAgent“. Dále vznikly automaticky reference na knihovny „AgentObjects a AxAgentObjects“.

Objekt „AxAgent1.Characters“ použije metodu „Load“ se dvěma parametry pro načtení souboru Microsoft Agent. První parametr nazve index pole, do kterého je postava načtena a druhý obsahuje cestu k souboru. V případě zadání pouze názvu souboru standardní cesta vede do „C:\\WINDOWS\\MSAGENTS\\CHARS“. Takto lze načíst do pole objektu libovolný počet postav. Avšak toto pole objektů nelze použít pro práci s agenty. Pro zjednodušení spustitelnosti aplikace jsou veškeré soubory potřebné pro aplikaci načítány z cesty „C:\\SOUBORY\\“.

Pro jednotlivé postavy nám slouží objekty instance třídy „IAgentCtlCharacter“ nacházející se v knihovně „AgentObjects“. Objekty postav bych doporučoval kvůli přehlednosti pojmenovávat podle načítaných postav. Do objektu postavy lze postavu jednoduše přiřadit znaménkem „=“ z daného indexu objektu „AxAgent1.Characters“. Od této doby všechny operace provádíme pouze tímto objektem.

V mé práci jsem pojmenoval objekt postavy „Sharp“. Použití objektu dále spočívá pouze ve volání metod „Show“, „Play“, „Speak“ a „Hide“.

5.3.2 Spuštění aplikace Microsoft PowerPoint

Pro použití aplikace PowerPoint v mé práci bylo nejprve nutno přidat reference na knihovny „PowerPoint“ a „Microsoft.Office.Core“. Dále jsem již v programu používal instanci tříd PowerPoint.Application a další. Pro druhou úlohu ve které jsem sestavoval snímky prezentace přímo programem bylo zapotřebí použití dalších tříd pro různá nastavení jako délka trvání snímku, fontu písma a další.

Takto lze nastavit prakticky cokoli. Jediné co se mi nepodařilo, zachytit události z aplikace PowerPoint do aplikace mnou vytvořenou.

V obou praktických úlohách jsem nejprve pomocí aplikace spustil prezentaci na celé okno a poté zobrazil Microsoft Agent. Ovšem na pořadí nezáleží. „PowerPoint“ programem lze spustit tak i vypnout a to s uložením nebo bez.

5.3.3 Použití Windows Media Player

„Windows Media Player“ jsem použil v práci na spouštění dubingu. Před použitím je potřeba přidat „Windows Media Player“ na „toolbox“ a následným přetažením ikony „Windows Media Player“ na formulář aplikace. Tímto jednoduchým způsobem lze přidat „Windows Media Player“ do aplikace. Při běhu programu aplikace přečte příkaz pro otevření souboru a přehrávač ihned začne přehrávat zvukovou stopu.

5.4 První úloha praktické části – popis

V první úloze byla řešena problematika použití Microsoft Agent jako podpora výuky či přednášky. „Microsoft Agent“ je nadabovaný a přednáška podporuje obsah školní prezentace „PowerPoint“ určené pro střední školy. Při spuštění aplikace se otevře formulář uveden na obrázku 11. Ovládací panel přesouvá prezentaci dopředu i zpět. Při přesunu se přehodí snímek prezentace, spustí se dubbing a postava zároveň používá i neverbální komunikaci. Jako doplňující informace jsou zobrazeny pomocí balónových textových dialogů.



Obrázek 11 – Ovládací panel první úlohy

Další tlačítko v aplikaci ve tvaru černého čtverce slouží k ukončení aplikace a zároveň k ukončení prezentace „PowerPoint“. Napravo od ovládacího panelu za použití „Labelu“ je zobrazen aktuální přehrávaný snímek. Použitá prezentace obsahuje 17 snímků a ke každému agent vypraví informace pomocí dubingu.

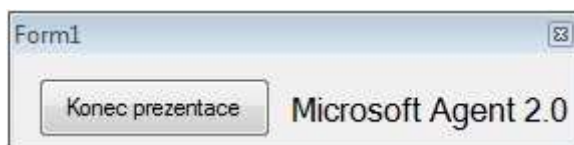
Aplikaci jsem navrhnul tak, aby byla uživatelsky přívětivá. Velmi jednoduchý design splňující zcela účel aplikace. Aplikace je určena pro formát obrazu 4:3 vzhledem k tomu, že je navržena k projekci. Na rozlišení obrazovky nezáleží. Program při spuštění zjistí rozlišení monitoru automaticky a tím zajistí správný pohyb agenta napříč po obrazovce. Agent se nepohybuje absolutně, ale relativně vzhledem k rozlišení.

Úkolem této úlohy je prakticky ukázat použití Microsoft Agent s pomocí dubingu při výuce či prezentace. Taktéž odzkoušet použití podpůrných aplikací jako je „Windows Media Player“ nebo „PowerPoint“.

5.5 Druhá úloha praktické části – popis

V této úloze jsem se zabýval vytvořením ukázkové prezentace přímo aplikací a synchronizací dvou Microsoft Agentů.

Při spuštění programu se vygeneruje prezentace „PowerPoint“ obsahující 4 krátké snímky. Tyto „snímky“ po ukončení programu nejsou uloženy. Obsah prezentace obsahuje informace o vytvoření programu a prezentace. Na obrázku 12 je uveden vzhled formuláře.



Obrázek 12 – Ovládací panel druhé úlohy

Po spuštění aplikace se okno aplikace PowerPoint rozprostře na celou obrazovku a snímky prezentace jsou automaticky spouštěny po 20 vteřinách.

Prezentaci doprovázejí dva agenti. První „Csharp“, kterého jsem animoval já a druhý „Peedy“ od společnosti „Microsoft“. Oba agenti doprovázejí aplikaci pomocí balónových dialogů a gest. Na konci prezentace se rozloučí a zmizí. Hlavním úkolem této úlohy bylo ukázat, jak lze pomocí programu vytvořit prezentaci a synchronizovat pohyby animací více Microsoft Agentů.

Obě praktické úlohy požadují operační systém „Windows Xp“ a „Microsoft Office 2003“.

6 Závěr

Cílem mého projektu bylo vytvořit animovanou postavu, která pomocí dubbingu a prezentace přednáší pro střední školy. Používaná prezentace je určena pro propagaci FEI. Dalším cílem mé práce bylo sestavení ukázkové prezentace za pomoci programového kódu. Ta je řízena dvěma vzájemně synchronizovanými Microsoft Agenty

V první řadě bylo třeba zvládnout animaci postavy reprezentující fakultu Elektrotechniky a Informatiky tak, aby co nejlépe vystihovala zaměření fakulty. Jednotlivé snímky byly tvořeny s pomocí „Adobe Photoshop 7.0 CE“ a následně ukládány do formátu 8 bitového BMP souboru. Takto importované snímky animace stačilo poskládat v aplikaci „Microsoft Agent Character Editor“. Importovaný soubor formátu „ACS“ je plnohodnotný „Microsoft Agent“ univerzálně použitelný např. v „Microsoft Office 2003“. Z časové náročnosti jsem splnil pouze požadovaný počet animací, ovšem kdyby bylo více času, šla by takto vytvořit celá řada animací, které by postavu tvořily přirozenější a rozmanitější.

Dalším krokem bylo sestavení první aplikace za pomoci „Visual C#“. Aplikace byla navržena tak, aby její ovládací panel přesouval snímky prezentace dopředu i dozadu. Tedy dle potřeby uživatele. Při přechodu snímků prezentace jsou spouštěny animace postavy a zároveň zvukové záznamy přes „Media Player“ určené k dubbingu. Tato metoda není pro dubbing nejvhodnější, ale pro daný účel vyhovující. U této úlohy jsem původně chtěl snímky prezentace přesouvat aplikací „PowerPoint“, moje aplikace by filtrovala události a na ně reagovala a spouštěla animace a dubbing. Bohužel tato metoda se nezdařila a pro posun snímků prezentace používám metodu „Sendkeys“, která přesune snímek prezentace na další či předchozí jako by ji stiskl uživatel přes klávesnici.

U druhé úlohy šlo o vytvoření krátké prezentace obsahující 4 snímky pod vedením dvou Microsoft Agentů, kteří se navzájem střídají v přednesu. Přednes využívá pouze dialogová balónová okna. Snímky jsou vytvářeny samotným programem, který po ukončení prezentace snímky neuloží. V obsahu prezentace jsou informace jak byla aplikace vytvořena. U této úlohy jsem ověřil synchronizaci Microsoft Agentů a také tvorbu prezentace s využitím knihoven „PowerPoint 2003“.

Obě úlohy byly vytvořeny ve vývojovém prostředí „Visual C#“ a byly v nich zahrnuty knihovny „Microsoft Agent Control 2.0“, „PowerPoint“ a „Windows Media Player“.

Výsledkem mé práce jsou 2 aplikace uživatelsky velmi jednoduché. Ze zkušeností při tvorbě aplikace jsem došel k závěru, že využití Microsoft Agentů je velmi univerzální a při vývoji mě napadala celá řada využití takto animovaných postav.

Literatura

- [1] NAGEL, Christian. C 2008 Programujeme profesionálně, Brno, Computer
- [2] Microsoft Agent home page [online]. 2009-05-01 [cit. 2010-06-20]. Dostupný z WWW: <<http://www.microsoft.com/products/msagent/main.aspx>>
- [3] Microsoft Developer Network – Microsoft Agent [online]. 2004 [cit. 2010-06-20]. Dostupný z WWW: <[http://msdn.microsoft.com/eu-us/library/ms695784\(VS.85\).aspx](http://msdn.microsoft.com/eu-us/library/ms695784(VS.85).aspx)>
- [4] Microsoft Developer Network – Assistant Object [online]. 2004 [cit. 2010-06-20]. Dostupný z WWW: <<http://msdn.microsoft.com/eu-us/library/aa432034.aspx>>
- [5] Microsoft Developer Network – Using the Microsoft Agent Character Editor [online]. 2004 [cit. 2010-06-20]. Dostupný z WWW: <[http://msdn.microsoft.com/eu-us/library/ms695797\(VS.85\).aspx](http://msdn.microsoft.com/eu-us/library/ms695797(VS.85).aspx)>
- [6] MASH - Microsoft Agent Scripting Helper [online]. 2008-03-10 [cit. 2010-06-20]. Dostupný z WWW: <<http://www.bellcraft.com/mash/>>
- [7] Microsoft Agent Ring [online]. 2008-01-14 [cit. 2010-06-20]. Dostupný z WWW: <<http://www.msagentring.org/>>
- [8] Phone book application – Introducing multimedia using Microsoft Agent [online]. 2004 [cit. 2010-05-15] Dostupný z WWW: <http://www.deitel.com/newsletter/20031222/simplycsp1_27.pdf>
- [9] How to use Microsoft Agent in C# [online] 2004-09-08 [cit. 2010-05-15]. Dostupný z WWW: <<http://www.codeproject.com/KB/cs/usemsagentcsharp.aspx>>
- [10] Adding a unique user interface to your scripts with Microsoft Agent technology [online] 2001-06-25 [cit. 2010-6-25]. Dostupný z WWW: <http://articles.techrepublic.com.com/5100-10878_11-1052487.html>
- [11] Způsob zpracování události PowerPoint s Visual C# .NET [online]. 2007-01-30 [cit. 2010-06-08]. Dostupný z WWW: <<http://support.microsoft.com/kb/308825>>
- [12] Embed PowerPoint Viewer in C# Win Form [online]. 2009-08-09 [cit. 2010-06-08]. Dostupný z WWW: <<http://stackoverflow.com/questions/1259369/embed-powerpoint-viewer-in-c-win-form>>
- [13] Způsob zpracování události PowerPoint pomocí Visual Basic .NET 2003 [online]. 2007-05-24 [cit. 2010-06-18]. Dostupný z WWW: <<http://support.microsoft.com/kb/824021/cs>>

[14] C# : Power point slide show in Windows Application [online]. 2006-2008 [cit. 2010-06-15]. Dostupný z WWW: <<http://www.eggheadcafe.com/tutorials/aspnet/86c1636b-0197-4d86-a452-e18866abd2cc/c--power-point-slide-sh.aspx>>

[15] C# HowTo symplex [online]. 2006-2008 [cit. 2010-06-15]. Dostupný z WWW: <<http://www.add-in-express.com/support/addin-c-sharp.php#powerpoint>>

[16] Using Windows Media Player with .NET [online]. 2003-11-01 [cit. 2010-06-28]. Dostupný z WWW: <http://translate.google.cz/?js=y&prev=_t&hl=cs&ie=UTF-8&layout=1&eotf=1&text=Bakal%C3%A1%C5%99sk%C3%A1+pr%C3%A1ce&file=&sl=cs&tl=en#en|cs|november>