

Vývojové prostriedky pre tvorbu MUME aplikácie

Portál: edu.ukf.sk - Vzdelávací portál - Univerzita
Konštantína Filozofa, Nitra

Kurz: Projektovanie multimedialnych aplikácií

Kniha: Vývojové prostriedky pre tvorbu MUME
aplikácie

Vytlačil(a): Zuzana Pavlendová

Dátum: Streda, 1 december 2021, 17:55

Opis

Vývojové prostriedky pre tvorbu MUME aplikácie

Obsah

1 Autorské systémy

2 MS PowerPoint

- 2.1 Vkladanie mediálnych elementov
- 2.2 Prechod snímok v aplikácii
- 2.3 Zabezpečenie interaktivity v MS PowerPoint
- 2.4 Integrácia flash animácií do programu MS PowerPoint

3 Authorware 7 Professional

4 Adobe FLASH ako optimálny prostriedok na tvorbu MUME aplikácií

- 4.1 Časová os (Time Line)
- 4.2 Základné symboly a ich umiestnenie v knižnici
- 4.3 Nástroje
- 4.4 Tvorba tlačidiel
- 4.5 Tvorba jednoduchých animácií
- 4.6 ActionScript v Adobe Flash

1 Autorské systémy

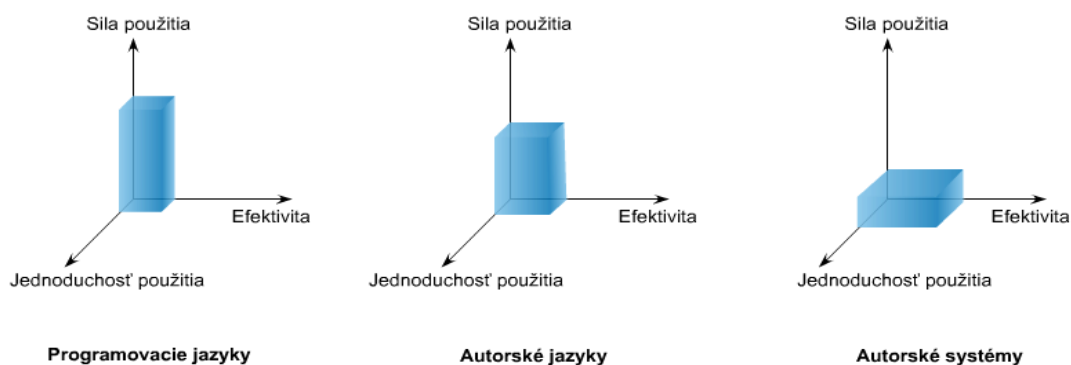
Autorské systémy sú profesionálne nástroje, určené pre tvorbu MUME aplikácií, s minimálnymi programátorskými nárokmi. Práca v nich je tak jednoduchá, že aj začínajúci používateľ bez znalostí z oblasti programovania môže vytvárať MUME aplikácie podľa svojich predstáv, avšak aj skúsenejší používatelia si prídu na svoje, keďže často je v takýchto systémoch zakomponovaný aj programátorský mód, umožňujúci im v systéme programovať podobne ako v bežných programovacích jazykoch a vytvárať tak komplikované a rozsiahle aplikácie.

Autorské systémy (AS) môžeme rozdeliť na:

1. *AS pracujúce na princípe slide show* – umožňujú vytvárať sekvenciu za sebou idúcich obrazoviek, ktoré sa môžu zobrazovať aj automaticky (bez zásahu používateľa – napr. MS PowerPoint).
2. *AS pracujúce na princípe hypertextu* – ich základom je správna štruktúra jednotlivých stránok. Stránka predstavuje uzol spojený s inými uzlami pomocou väzieb, ktoré definuje autor podľa významových súvislostí. Takéto hypertextové systémy dovoľujú okrem textu pracovať aj s grafikou a využívať nezávislé externé programy. Typickými predstaviteľmi sú ToolBook alebo HyperCard.
3. *AS pracujúce na princípe objektovo - orientovaného programovania*. Najznámejším predstaviteľom takéhoto AS je Authorware 7 Professional od firmy Macromedia (v súčasnosti už Adobe) a IconAuthor.

Každý program vytvorený pomocou určitého programovacieho jazyka je spracovávaný pomocou interpretácie alebo kompilácie.

Podobne spracovávajú autorské systémy aj MUME aplikáciu. Pri interpretácii programu sa postupne vykonávajú všetky príkazy, pričom môžeme začať okamžite vykonávať ktorúkoľvek časť programu, čo značne urýchľuje jeho tvorbu. Pri kompilácii sa najskôr preloží zdrojový text do strojového kódu procesora a až potom je možné program spustiť, čo je často časovo veľmi náročné.



Obr.1 Porovnanie nástrojov pre tvorbu MUME aplikácií

K tvorbe kvalitných MUME aplikácií sú však vhodnejšie autorské jazyky, ktoré odstraňujú nedostatky AS tak, že zvyšujú silu ich použitia a poskytujú zároveň dostatočnú efektivitu. Aj preto sa v tejto publikácii zameriame skôr na softvérový produkt Adobe Flash, ktorý je v súčasnosti štandardom pri tvorbe aplikácií a predstavuje kombináciu autorských jazykov a autorských systémov. Pri výbere vhodného AS je potrebné zvážiť, načo ho chceme využívať. Ak chceme vytvárať iba jednoduché prezentácie s obmedzenými možnosťami interaktivity, môžeme použiť MS PowerPoint, ale ak chceme vytvoriť aplikáciu s možnosťou testovania a vyhodnocovania správnych odpovedí, je výhodnejšie použiť AS Authorware Professional.

2 MS PowerPoint

Jedným zo základných autorských systémov určený na tvorbu MUME aplikácií, bez nutnosti používateľa vedieť programovať, je MS PowerPoint. Aj mierne pokročilý používateľ vie veľmi ľahko vytvoriť či upraviť už vytvorenú aplikáciu podľa svojich predstáv a aktuálnych potrieb. K pokročilejším zručnostiam pri tvorbe pomocou MS PowerPoint patrí vlastná tvorba nelineárnej multimediálnej aplikácie s použitím makier. Po spustení MS PowerPoint sa dostaneme do prostredia, ktoré je podobné, aké poznáme aj z iných programov balíka MS Office. Ikony na paneli nástrojov a paneli na formátovanie, pripomínajú prostredie MS Word, ale tieto sú doplnené o niektoré špecifické nástroje, ktoré sú typické pre vytváranie MUME aplikácií a jednoduchých prezentácií.



Vzhľad MS PowerPoint

Obr.2 Vzhľad MS PowerPoint

Po spustení MS PowerPoint môžeme použiť na vytvorenie novej aplikácie dialógové okno, v ktorom si zvolíme, či chceme otvoriť čistú prezentáciu, to znamená, že všetky prvky (snímky, nadpisy, obrázky...) a formáty (veľkosť, štýl, farbu pozadia atď.) vytvoríme ručne, alebo použijeme Sprievodcu, či niektorú z preddefinovaných šablón. Ak sa necháme inšpirovať ponukou vopred vytvorených šablón, môžeme ich kedykoľvek upraviť podľa svojej potreby. Priebeh tvorby aplikácie sa zobrazuje v Pracovnej ploche. V ľavej časti sa nachádza zoznam jednotlivých snímok a náhľad ich obsahu. Dialógový riadok v spodnej časti umožňuje meniť spôsob zobrazovania snímok a spúšťať samotnú prezentáciu. V spodnej časti obrazovky sa nachádza *časť pre poznámky*, kde je preddefinovaný text „*Kliknite sem a zadajte poznámky*“ (obr. 59). Tu je možné vložiť poznámky k jednotlivým snímkam prezentácie. Práca s programom MS PowerPoint sa veľmi podobá práci s MS Word. Zmena typu, veľkosti písma alebo zarovnania, sa vykonáva rovnako ako v MS Word. Aj preto môžeme veľmi jednoduchým spôsobom podľa potreby vkladať text, obrázok alebo iný objekt. Ak potrebujeme vložiť nový prvok, použijeme ponuku *Vložiť*. MUME aplikácia vytváraná pomocou MS PowerPoint pozostáva zo snímok a každá snímka pozostáva zo samostatne osadených častí – blokov. Výhoda snímok a blokov spočíva v možnosti nastavovať efekty prechodu a ich časovanie, resp. ich spúšťanie napr. kliknutím myšou. Každý efekt, ktorý použijeme, je možné pozorovať vďaka náhľadu a preto aj začiatovník dokáže dané ovládacie prvky nastaviť. Vloženie novej snímky vykonáme príkazom *Nová snímka* zo záložky *Domov*. Snímky môžeme odstraňovať, presúvať alebo kopírovať.

Kvalitná grafická úprava má pri tvorbe MUME aplikácie svoje nezastupiteľné miesto. Grafický vzhľad aplikácie vytvárame zmenou grafických nastavení objektov snímok, resp. vkladáním samotných objektov.



Formátovanie pozadia

Obr.3 Formátovanie pozadia

Zmenu pozadia zvolíme pomocou voľby *Štýly pozadia>Formátovať pozadie* v záložke *Návrh* alebo pomocou pravého tlačidla myši a voľby *Formátovať pozadie*. Pozadie snímky môžeme použiť na jednu aktuálnu alebo na všetky snímky budúcej aplikácie.

2.1 Vkládanie mediálnych elementov

Text

Texty, ktoré vkladáme do MUME aplikácie sú umiestnené v zástupných symboloch, nazývaných textové polia. Ich polohu a veľkosť môžeme podľa potreby meniť pomocou myši, alebo priamym zadáním hodnôt. Pre lepšiu prehľadnosť a orientáciu môžeme upraviť veľkosť, tvar, typ, farbu písma.



Obr.4 Panel nástrojov Písmo

Úpravu textu v textových poliach zarovnávaním vykonávame rovnako ako v textovom editore - napravo, naľavo, vycentrované v strede, zarovnané do bloku (podľa okrajov).



Obr.5 Panel nástrojov Odsek

Zarovnanie sa realizuje pomocou tlačidiel v bloku Odsek a výberom konkrétneho zarovnaní.

Textové pole, ktoré nepotrebujeme, môžeme vymazať jednoduchým stlačením klávesy *Delete* alebo *Backspace*.

Obrázok

Okrem textov môžeme ako ďalší multimediálny element vkladať obrázky pomocou záložky *Vložit>Obrázok* (ClipArt). Prípadnú úpravu ako formát, jas alebo kontrast vykonávame cez príkaz *Formátovať obrázok* (po kliknutí pravého tlačidla myši na obrázok).



Obr.6 Panel nástrojov Ilustrácie

Zvuk

To, čo môže byť v MUME aplikácií nositeľom informácie alebo jednoduchým sprievodným efektom pri vzájomnom kontakte hypertextového odkazu a myši je zvuk. Zvuk je možné priradiť:

- akémukoľvek objektu na snímke (nakreslené tvary, tlačidlá, a pod.),
- akémukoľvek animačnému efektu na snímke,
- samotnej snímke pri prechode na novú snímku
- pozdĺž celej prezentácie ako hudobný podmaz.

Zvuk vkladáme pomocou galérie médií alebo ako externý súbor. Súčasťou galérie médií pri typickej inštalácii MS PowerPoint sú jednoduché zvuky uložené vo formáte WAV. Po vložení zvuku z galérie sa objaví na snímke malá ikona reproduktora.

V normálnom zobrazení je možné ju skryť, zmeniť jej veľkosť, či presunúť. Pri používaní MUME aplikácie sa zvuk prehráva automaticky alebo pri kliknutí na ňu. Pre zvuk vložený zo súboru platia rovnaké pravidlá pre úpravy ako pre zvuk z galérie médií.

Video

Video ako jeden zo základných multimediálnych elementov je nasnímaná sekvencia obrázkov, ktorá môže MUME aplikáciu „oživiť“, alebo doplniť o informácie podané formou pohyblivého obrazu. Videá používané pri tvorbe

aplikácií v MS PowerPoint delíme do týchto základných kategórií:

- videá, ktoré boli zaznamenané v reálnom svete, napr. digitálnou videokamerou,
- videá, ktoré boli zaznamenané v počítači pomocou softvéru (Screen Capture)
- animované videá – sú to animácie vytvorené v špeciálnych programoch napr. 3D Studio Max alebo Adobe Flash a pod.,

Video vkladáme podobne ako zvuk a to buď z Galérie médií alebo zo súboru. Spôsob prehrávania môžeme usmerniť pomocou jednoduchého nastavenia niekoľkých parametrov. Vložené videá je možné spúšťať buď automaticky alebo po klepnutí myši na ikonu.

Animácie a animačné efekty

Animáciami jednotlivých objektov môžeme riadiť sled prijímaných informácií a tak následne upriamiť pozornosť používateľa MUME aplikácie k aktuálnemu problému, čo zabraňuje pocitu zo zahltenia množstva informácií na aktuálnej snímke.

Efekty animácie pridávajú pohyb textu, obrázkom a ďalšiemu obsahu aplikácie. Okrem pridania akcie umožňujú upútať pozornosť, zdôrazniť dôležité body, vytvárať prechody medzi snímkami a maximalizovať priestor snímky premiestňovaním obsahu. Ak sa použijú správne, obohatia MUME aplikáciu o prekvapenie. Pomocou funkcií animácie v programe MS PowerPoint vytvoríme také efekty pre text a obrázky, ako je prílet, odrážanie, vzdďalovanie a ďalšie.



Animovanie objektu

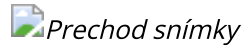
Obr.7 Animovanie objektu

Najjednoduchší spôsob pridávania animácií, ktoré majú používať preddefinovanú alebo pripravenú schému animácie, vyvoláme pomocou príkazu v záložke *Animácie*.

Schéma predstavuje skupinu vizuálne prítťažlivých efektov, ktoré možno niekoľkými kliknutiami myšou použiť v určitých snímkach alebo v celej aplikácii. Šetrí čas a umožňuje rýchlo dosiahnuť požadované výsledky. Schéma predstavuje aj určitý spôsob zachovania konzistentnosti, pretože na každej snímke sa používajú tie isté efekty. Schéma okrem toho nie je pevne daná – môžeme ju jednoducho odstrániť, nahradiť alebo prispôbiť.

2.2 Prechod snímok v aplikácii

Animačným a zvukovým efektom určeným pre celé snímky, hovoríme *prechody snímok*. Prechodové efekty pre striedanie snímok môžeme nastaviť v záložke *Animácie*.



Obr.8 Prechod snímky

Nastavenie rýchlosti a zvuku prechodu

Prechodové efekty snímok sa pri vytvorení automaticky nastavujú na rýchly priebeh a nebývajú sprevádzané zvukom. Rýchlosť ich predvádzania i prípadný zvukový sprievod však môžeme jednoducho vytvoriť. Rýchlosť s akou vystrieda nová snímka predchádzajúcu, nastavíme vhodnou voľbou v záložke *Animácie*. Možnosti sú: pomaly, stredne, rýchlo.



Obr.9 Rýchlosť prechodu snímky

Nastavenie zvukového efektu prechodu

V záložke *Animácie* si môžeme vybrať aj zvukový efekt, ktorý bude zobrazovanie snímok sprevádzať. Označením voľby *Cyklicky o pakovať do ďalšieho zvuku* zaistíme jeho opakované prehrávanie, pokiaľ nezaznie iný zvuk. Voľbou *Bez zvuku* prehrávanie skôr nastaveného zvuku ako súčasť príchodu snímky zrušíme. Voľba *Zastaviť predchádzajúci zvuk* dokonca zaistí, že s príchodom snímky ustane prípadný zvuk znejúci už z niektorej predchádzajúcej snímky.



Obr.10 Zvuk prechodu snímky

Spúšťanie prechodu snímok

Zobrazovanie jednotlivých snímok, či už majú nastavené prechodové efekty alebo nie, môžeme pri projekcii ovládať ručne alebo nechať vykonávať automaticky. Môžeme tak určiť, či má byť daná snímka vystriedaná pri projekcii klepnutím myši alebo automaticky po určitom počte sekúnd. Druhý spôsob sa hodí pre automatické projekcie bez prítomnosti rečníka. Je možné nastaviť i obe možnosti súčasne. Potom sa ďalšia snímka objaví podľa toho, k čomu z týchto dvoch možností príde skôr.

2.3 Zabezpečenie interaktivity v MS PowerPoint

Podľa toho, ako sme si charakterizovali pojem interaktivita vieme, že jej zabezpečenie v MS PowerPoint je značne problematické. Interaktívnu aplikáciu musíme tak zostaviť veľmi prehľadne a logicky, aby používateľ mohol sám pristupovať k údajom, ktoré sú pre neho potrebné.

Veľký dôraz sa kladie najmä na správne umiestnenie ovládacích prvkov v aplikácii. Interaktívne prvky v MUME aplikácii vytvorenej pomocou MS PowerPoint pozostávajú najmä z tlačidiel a hypertextových odkazov.

Používať takéto rozdelenie je však veľmi všeobecné, keďže v praxi funkcia jedného alebo druhého prvku často splyva do jednej a tej istej veci, ktorou je nadefinovanie akcie alebo odkazu. Rozdiel je najmä v základnom vnímaní pojmu hypertextový odkaz, ako označenia pre činnosť vyvolanú kliknutím na text.

Pri tlačidle však akciu vyvoláme kliknutím na grafický objekt – zástupcu. Hypertextový odkaz sa prejaví farebným podčiarknutím, pri spustení, po klepnutí alebo pri prechode myšou nad odkazom. MS PowerPoint vytvára hypertextové odkazy s podčiarknutím automaticky, pri každom zápise e-mailovej adresy alebo webovej adresy URL do snímky.

Pri vytvorení hypertextového odkazu postupujeme nasledovne:

1. Vyberieme text alebo objekt, ktorému chceme hypertextový odkaz priradiť.
2. Príkazom *Hypertextový odkaz* (Prepojenie) v záložke *Vložiť* otvoríme dialógové okno *Vložiť hypertextový odkaz*. Tu do textového poľa *Adresa* napíšeme alebo vyberieme cieľ odkazu.

Okrem hypertextových odkazov môžeme použiť na zabezpečenie interaktivity aj už spomenuté tlačidlá. V prípade potreby nemusíme vždy vyrábať nové, ale môžeme použiť tie, ktoré sú implementované priamo v MS PowerPoint. Tlačidlá s akciou vložíme do snímky rovnakým spôsobom ako automatické tvary:

1. Na paneli nástrojov *Kreslenie* potvrdíme tlačidlo *Automatické tvary* a z ponuky vyberieme *Tlačidlá akcií*.
2. Po zvolení potrebného tlačidla sa presunieme na snímku, kde táháním myši určíme jeho tvar a veľkosť.
3. Automaticky sa zobrazí dialóg *Nastavenie akcie*, v ktorom sa už nachádza preddefinovaná akcia.
4. Pre zadefinovanie vlastnej akcie odstránime pôvodný text.

V prípade vytvorenia kvalitnej hypertextovej štruktúry pomocou odkazov alebo tlačidiel akcií, určíme v interaktívnej MUME aplikácii sled jednotlivých snímok a teda nemá ju zmysel dopĺňať nahratým hovoreným komentárom. Prvky interaktivity tak môžeme dodať akémukoľvek objektu s výnimkou pozadia snímky.

2.4 Integrácia flash animácií do programu MS PowerPoint

Ak sme pomocou programu napr. Adobe Flash alebo Adobe Captivate vytvorili animáciu a uložili sme ju ako súbor s príponou SWF, môžeme ju v prípade potreby prehrať aj v programe MS PowerPoint. Musíme však do vytvárannej prezentácie či aplikácie vložiť príslušný ovládací prvok ActiveX a vytvoriť z neho prepojenie na súbor s príponou SWF.

Ovládacie prvky ActiveX sa používajú na webových lokalitách a v aplikáciách v počítači, pričom nepredstavujú samostatné riešenia. Môžu byť spustené v programoch ako Internet Explorer alebo programoch balíka MS Office. Aj keď to možno na prvý pohľad nevyzerá, sú veľmi účinné, keďže ide o objekty typu COM (Component Object Model), ktorý využíva neobmedzený prístup k počítaču.

Ak nás však práve uvedená skutočnosť neodradila od nášho zámeru a predsa chceme vložiť animáciu takýmto spôsobom, musíme v MS PowerPoint zobraziť záložku *Vývojár*, cez *Možnosti programu PowerPoint*. Potom v bloku *Ovládacie prvky* karty *Vývojár* klikneme na položku *Ďalšie ovládacie prvky* (je to tlačidlo s ikonou kladiva a maticového kľúča).



Obr.11 Ovládacie prvky

Z ponuky ďalej vyberieme *Shockwave Flash Object*. Kurzorom myši naznačíme na ploche veľkosť nášho zobrazovaného Flash objektu. Klikneme naň pravým tlačidlom myši a vyberieme z kontextového menu položku *Vlastnosti*. Do položky *Movie* na karte *Alphabetic* zadáme relatívnu cestu objektu napr. *img\anim.swf* (obr. 69). Týmto krokom zabezpečíme načítanie objektu do programu MS PowerPoint. Adresár „img“ je umiestnený v hlavnom adresári, kde sa nachádza aj naša prezentácia, pričom v adresári img sa nachádzajú naše flash animácie.



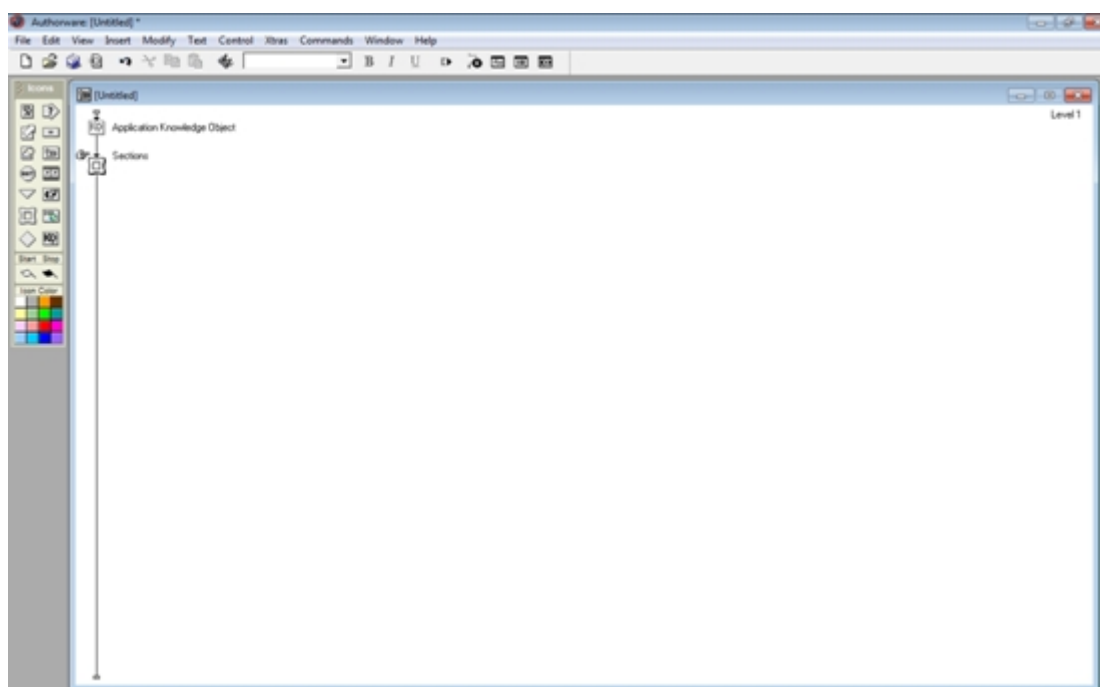
Obr.12 Vloženie flash animácie do MS PowerPoint

Druhou možnosťou ako vložiť animáciu do prezentácie programu MS PowerPoint je použitie špeciálnych programov, ktorých je však našťastie dostatok a sú navyše aj zadarmo. Jedným z nich je napr. *Swiff Point Player*, ktorý integruje voľný plugin do prezentácie vytvorenej pomocou MS PowerPoint, pričom na rozdiel od ActiveX nevidíme iba prázdny „kríž“ ale aj náhľad samotnej prezentácie.

3 Authorware 7 Professional

Authorware Professional (AP) je komplexný systém vytvorený spoločnosťou Macromedia, dnes už Adobe, ktorý dokáže veľmi jednoduchým spôsobom zvýšiť efektívnosť tvorby interaktívnych MUME aplikácií. Celý systém používa objektovo orientované prostredie založené na ikonách, ktoré sa vkladajú do časového diagramu. Rovnako ako aj v prípade produktu Flash, tiež je možné využívať výhody skriptovacieho jazyka veľmi podobného JavaScriptu a použiť možnosť exportovania do súboru s príponou EXE, ktorý spustíme na ktoromkoľvek počítači, bez potreby dodatočnej inštalácie AP alebo jeho časti. Authorware Professional taktiež umožňuje import a export súborov typu XML, čo môžeme využiť napr. pri importe prezentácie vytvorenej v MS PowerPoint a následne ju ďalej upravovať už v autorskom systéme.

Po spustení programu a otvorení nového súboru máme k dispozícii pracovné prostredie, kde je graficky na diagrame vyznačený začiatok a koniec programu.



Obr.13 Authorware 7 Professional

Pomocou presúvania ikon z panela nástrojov nachádzajúceho sa v ľavej časti obrazovky na diagram, môžeme vytvárať jednoduchú blokovú schému programu. Každá z ikon má určitý špecifický význam, môžeme ju otvoriť a v prípade potreby aj zeditovať.

V súčasnosti sa o ďalšom vývoji tohto produktu do budúcnosti neuvažuje, keďže ho plne nahradil Adobe Flash a iné podporné produkty pre tvorbu multimediálnych aplikácií.

4 Adobe FLASH ako optimálny prostriedok na tvorbu MUME aplikácií

Multimediálna aplikácia umožňuje interaktívne prepojenie jej základných elementov ako sú: text, obraz, video, animácia a zvuk. Preto je veľmi dôležité vybrať si na jej tvorbu ten správny program, keďže v súčasnosti je ich na trhu viac než dost. Otázne však je, či spĺňajú našu predstavu o vytvorení kvalitnej MUME aplikácie. Na základnú tvorbu môžeme použiť aj program Microsoft PowerPoint, ale takto vytvorená aplikácia by bola však skôr prezentáciou ako multimediálnou aplikáciou, umožňujúcou interaktívne pôsobenie napr. pri animáciách.

Na tvorbu multimediálnych aplikácií sú vhodné také programy, ktoré umožňujú pracovať so všetkými základnými elementmi MUME aplikácie. Jedným z takýchto programov je aj grafický program Adobe Flash.

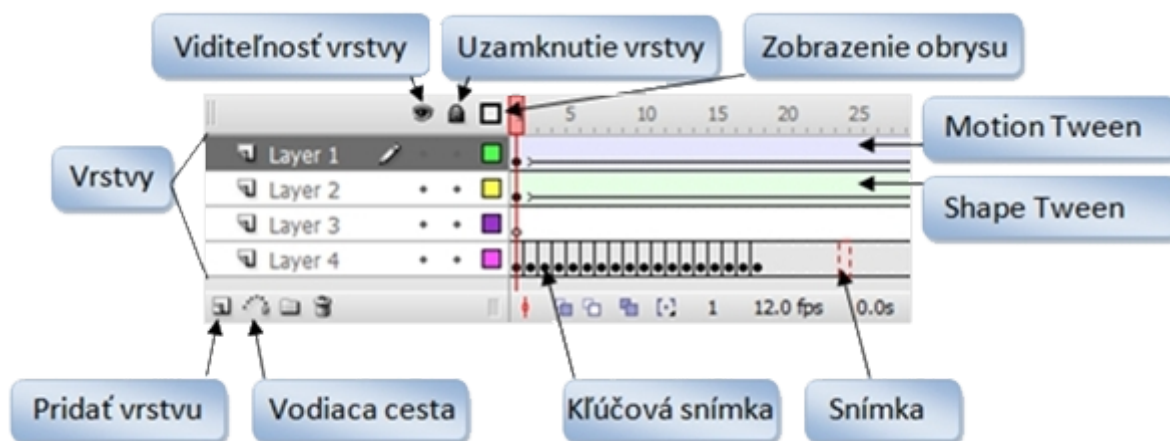
Najväčšia výhoda tvorby MUME aplikácií v Adobe Flash je v možnosti implementovania prvkov interaktivity do aplikácií, ako aj v možnosti importovania zvukov, pri maximálnej kompresii podobne, ako sme na to zvyknutí u súborov MP3. Podobne ako v Adobe Photoshop alebo v CorelDRAW pracujeme s vrstvami (Layers), ktoré sa najviac využívajú pri animáciách. Program je výnimočný v jednoduchosti, akou môžeme vytvoriť interaktívne animácie s minimálnymi znalosťami z oblasti programovania. Preto môžeme povedať, že kombinuje výhody autorských systémov a autorských jazykov, zvyšuje „mocnosť“ - silu ich použitia. Tí, ktorí už dlhšie pracujú s Adobe Flash, nám isto dajú za pravdu, že je to naozaj univerzálny prostriedok. Umožňuje kresliť aj importovať grafické súbory, v prípade potreby z bitmapy vytvárať vektorový obrázok pomocou príkazu *Trace*. Ten, kto sa v tomto grafickom programe naučí profesionálne pracovať, už pravdepodobne nebude chcieť používať iný softvér. Preto je potrebné si uvedomiť, že Adobe Flash je síce naozaj veľmi silný grafický nástroj, ale pre kvalitnú tvorbu MUME aplikácií budeme musieť kombinovať rôzne techniky práce pomocou rôznych softvérových produktov.

Od verzie Flash 5 je používateľské prostredie štandardné. Adobe Flash podobne ako aj iné programy používané v súčasnosti, má uniformný dizajn a preto aj jeho menu je veľmi podobné tým, ktoré poznáme z programov pracujúcich pod OS Windows.

V prípade, že by sme chceli predstaviť prostredie a vysvetliť prácu v Adobe Flash, museli by sme siahnuť po publikáciách na to špeciálne určených. Na doplnenie si poznatkov z tejto oblasti je však možné použiť rôzne web portály, akými sú napr. www.flash.cz, www.inspired.sk alebo www.flashland.sk. K práci v programe Adobe Flash sa vrátíme ešte v poslednej kapitole tejto publikácie, kde si na konkrétnom príklade tvorby MUME aplikácie ukážeme základné techniky práce, ako aj tvorbu ovládacích tlačidiel, interaktívnych animácií a ďalšie postupy.

4.1 Časová os (Time Line)

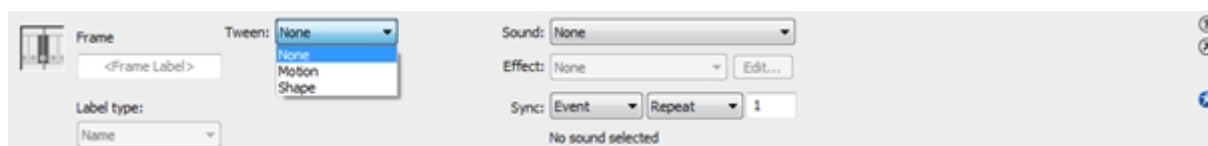
Slúži na vytváranie a ovládanie časového priebehu MUME aplikácie, podobne ako aj v iných autorských systémoch, kde sa používa napr. časový diagram. Základným prvkami časovej osi sú: *frame* (snímka, F5) a *keyframe* (klúčová snímka, F6).



Obr.14 Časová os

V prípade vytvorenia animácie nemusíme vytvárať vždy nové kľúčové snímky, ale postačí ak zadáme začiatočnú a koncovú polohu objektu, keďže Adobe Flash umožňuje tzv. *dopočet zvyšných snímok*. O tom, aký *dopočet snímok* sme použili nás informuje farebné označenie na časovej osi. Poznáme dva základné spôsoby *dopočítavania snímok* a to pomocou:

1. *Tweening* – *dopočítavanie pohybu*,
2. *Shape* – *dopočítavanie zmeny tvaru objektu* (napr. keď z kruhu vznikne obdĺžnik).



Obr.15 Spôsob *dopočítavania snímok* na paneli *Properties*

4.2 Základné symboly a ich umiestnenie v knižnici

Symbole *Sound* a *Bitmap* sa v Adobe Flash vytvoriť nedajú, pretože predstavujú importované súbory. Konkrétne u *Sound* je to ľubovoľná vložená zvuková stopa.

Podobne je to aj so symbolom *Bitmap*, ibaže ide o importované grafické súbory (napr. JPG, GIF a pod). Symbole *MovieClip*, *Button* a *Graphic* sa vytvárajú priamo v Adobe Flash pomocou príkazu *Insert > Convert to Symbol* (F8).

Pri práci v Adobe Flash používame 3 základné druhy symbolov:

a) *Graphic* - Najväčšie využitie má pri statickej grafike, ale dajú sa v ňom vytvárať aj animácie.

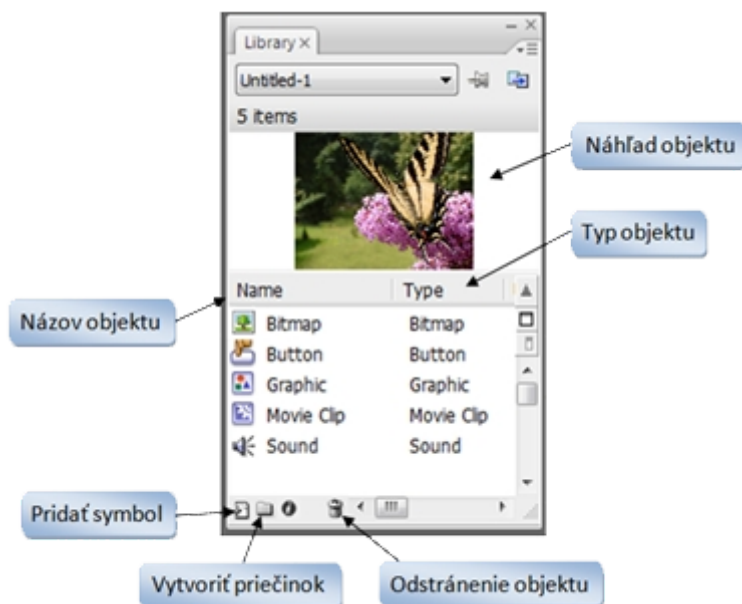
b) *MovieClip* - Je primárne určený na tvorbu animácií a programovanie v symboloch. Od hlavnej scény sa chová absolútne autonómne. Je to vlastne scéna v scéne, v ktorej fungujú všetky príkazy typu ActionScript. Výhodou tohto symbolu oproti symbolu *Graphic* je možnosť inštancie – mena, pomocou ktorého budeme *MovieClip* ovládať. Môžeme tak ovplyvňovať ľubovoľné vlastnosti (napr. veľkosť, viditeľnosť alebo polohu), po splnení určitých podmienok definovaných v ActionScripte.

c) *Button* - Symbol, ktorý sa chová ako tlačidlo. Pri editácii symbolu sa otvorí špeciálne upravená časová os, na ktorej sa nachádzajú tzv. udalosti. Tlačidlá sú špecifická skupina, ktorá môže dodať celkový vzhľad multimedialnej aplikácii, ale môže aj naopak, poukázať jej celkový dojem. Rozdeľujeme ich do troch základných kategórii:

1. *statické* – nevykonávajú žiadny pohyb ale môžu meniť napr. farbu
2. *dynamické* – pri interakcii nastane pohyb alebo animácia tlačidla
3. tlačidlá statické alebo dynamické v *spojení so zvukom*

Všetky symboly, ktoré vytvárame v Adobe Flash alebo ich do neho importujeme, sa ukladajú do knižnice (Library, CTRL+L). Má podobnú funkciu ako napr. bežná (domáca) knižnica, ktorá obsahuje množstvo kníh. V prípade potreby si knihu vyberieme, prečítame a vložíme späť.

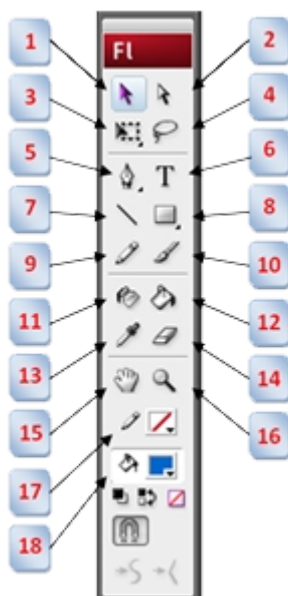
Ak potrebujeme s určitým objektom nachádzajúcim sa v knižnici pracovať na scéne, stačí ak ho vložíme do scény. Takto môžeme vložiť ľubovoľný počet objektov jedného druhu a samostatne im priradiť akciu. Je potrebné si však uvedomiť, že ak vykonáme akúkoľvek dodatočnú úpravu objektu na scéne, vykoná sa aj úprava objektu, ktorý je uložený priamo v knižnici.



4.3 Nástroje

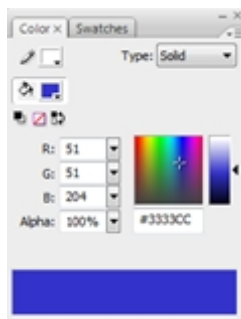
Panel *Tools* (*Nástroje*), ktorý slúži na kreslenie a editovanie objektov, sa štandardne nachádza v ľavej časti obrazovky.

- | | |
|------------------------------------|---|
| 1 Označenie a posúvanie objektov | 10 Kreslenie štetcom |
| 2 Základná editácia objektov | 11 Zafarbenie okrajov objektov |
| 3 Transformácia objektov | 12 Vypĺňanie oblasti |
| 4 Označenie plochy lasom | 13 Zistí aktuálnu farbu výplne a obrysu |
| 5 Vytváranie zaoblených čiar perom | 14 Gumuje – zmaže výplň a obrys |
| 6 Nástroj pre prácu s textom | 15 Navigovanie pomocou posúvania |
| 7 Kreslenie čiar | 16 Priblíženie a oddialenie plochy |
| 8 Vytváranie uholníkov a oválov | 17 Nastavenie farby obrysu |
| 9 Kreslenie ceruzkou | 18 Nastavenie farby výplne |



Obr.17 Panel nástrojov

Nástroj *Color Mixer* (Paleta farieb) slúži na rýchlu voľbu požadovanej farby ohraničenia alebo farby výplne objektu a stupňa jeho priehľadnosti (Alpha). Farbu výplne je možné nastaviť aj pomocou lineárneho alebo radiálneho gradientu.



Obr.18 Paleta farieb

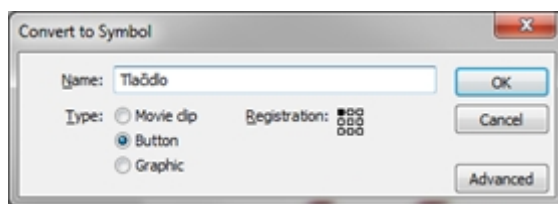
4.4 Tvorba tlačidiel

Existuje niekoľko možností, ako môžeme v Adobe Flash vytvoriť tlačidlo. Konkrétne zvolený postup riešenia závisí vždy od toho, načo bude určené. Ako príklad tvorby tlačidiel môžeme uviesť vytvorenie statického tlačidla z textu. Najskôr si označíme text, ktorý má byť tlačidlom, alebo hypertextovým odkazom. Stlačením CTRL+B (*Modify/Break Apart*) ho upravíme na krivky. Tento krok je potrebný, aby sa text zobrazoval rovnako na každom počítači.



Obr.19 Text upravený na krivky

Označený text skonvertujeme na typ *Button* pomocou klávesy F8. Pomenujeme ho, aby sme vedeli, pod akým názvom sa bude nachádzať v knižnici (CTRL+L) pre ďalšie použitie.



Obr.20 Vytvorenie symbolu Button z textu

Takto vytvorené tlačidlo je však potrebné ešte upraviť. Pri testovaní pomocou CTRL+Enter (Test Movie) je vidieť, že pri prechode kurzorom nad tlačidlom, nebola obsiahnutá celá plocha tlačidla, ale iba jednotlivé časti textu. Preto je potrebné vytvoriť tzv. podklad tlačidla. Pomocou dvojkliku na tlačidlo sa presunieme do plochy určenej pre jeho editáciu. Kliknutím pravého tlačidla myši na *Layer 1* otvoríme kontextové menu a vyberieme *Insert Layer*. Týmto krokom vložíme novú vrstvu potrebnú pre podklad tlačidla. Novú vrstvu *Layer 2* však musíme premiestniť pod *Layer 1*, inak by neslúžila ako podklad pre tlačidlo.



Obr.21 Vrstvy tlačidla

Vrchnú vrstvu uzamkneme, aby sme sa nepomýlili a náhodou do nej nezapisovali. Do druhej vrstvy vytvoríme podklad pre text napr. pomocou nástroja *Rectangle Tool*. V tomto prípade nie je dôležité, či náš podklad bude alebo nebude obsahovať okraje. Pokiaľ sme nezmenili medzičasom farbu, podklad nám prekryje aj náš text. Text sa však vôbec nezmení, keďže sme predtým uzavreli vrstvu, v ktorej sa nachádzal. Posledný krok, ktorý musíme vykonať k úspešnému ukončeniu tvorby tlačidla je nastaviť priehľadnosť pozadia tlačidla a tak získať podklad, ktorý je neviditeľný. Označíme si preto podklad a v paneli *Color Mixer* nastavíme položku *Alpha* na 0 %. Dvojklikom mimo plochy tlačidla alebo prepnutím sa na položku

Scene 1 sa dostaneme z plochy určenej na editáciu tlačidla. Takto vytvorené tlačidlo je pripravené na použitie, v prípade potreby môžeme upraviť aj jeho prechod. Ak teda chceme, aby sa pri prechode kurzoru nad tlačidlom menil text (napr. font, výška textu alebo farba), použijeme pri editácii udalosť *Over*.

Udalosti tlačidla:

UP - aktivuje sa, ak je myš mimo tlačidla,

OVER - aktivuje sa, ak je myš nad tlačidlom,

DOWN - aktivuje sa, ak je stlačené ľavé tlačidlo myši nad tlačidlom,

HIT - plocha, ktorou sa označuje aktívna oblasť tlačidla. Ak sa nachádza v tejto ploche nejaký tvar, pričom *UP*, *OVER* a *DOWN* sú prázdne, tlačidlo sa bude javiť ako neviditeľné (ale funkčné). Akcia pre dané tlačidlo sa vykoná iba vtedy, ak bola splnená podmienka v ActionScripte.

Pri tvorbe statického tlačidla, ktoré mení farbu alebo veľkosť postupujeme podobne. Pri dynamickom tlačidle môžeme použiť napr. vopred vytvorenú animáciu.

Pri projektovaní multimedialných aplikácií je jedným zo základných pravidiel aj použitie zvuku. Najčastejšie sa zvuky používajú pri výklade určitej problematiky (výučbový softvér) alebo pri interakcii s príslušným tlačidlom. Ak už máme vytvorené tlačidlo, je potrebné, aby sme pracovali v ploche určenej na jeho editáciu. Pokiaľ sme importovali zvuk do knižnice (ak nie, tak zadáme *File> Import to Library*), presunieme ho na udalosť *Over* alebo *Down* podľa toho, či sa má zvuk prehrávať pri prechode kurzorom nad tlačidlom alebo až pri jeho stlačení. Pri stlačení udalostí *Over* alebo *Down*, podľa toho kde sa nachádza vložený zvuk, môžeme v paneli *Properties* nastaviť jeho vlastnosti. Položka *Effect* je určená na prácu s kanálovým výstupom zvukovej karty. V položke *Sync.* (synchronizácia) nastavujeme akciu pre *Štart* alebo *Stop*, kedy sa má zvuk prehrávať. Vedľa sa nachádza položka *Loop* (slučka), ktorá určuje koľkokrát sa prehrá zvuk podľa nastaveného čísla.

4.5 Tvorba jednoduchých animácií

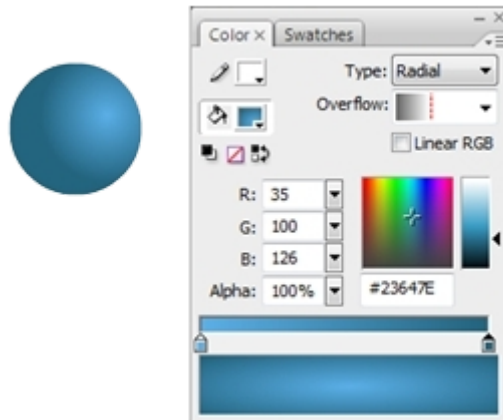
Pri multimediálnych aplikáciách predstavuje animácia jeden z jej základných elementov. Pokiaľ je vytvorená s možnosťou jej riadenia, obohacuje ju tak o ďalší interaktívny multimediálny prvok. Animácia môže byť vytvorená dvoma základnými spôsobmi:

1. snímka za snímkou
2. prvá a posledná kľúčová snímka, pričom ostatné snímky dopočíta za nás program.

Pokiaľ uvažujeme o prvej možnosti, museli by sme do každej snímky vkladať napr. obrázok, ktorý by sa postupne menil. Táto metóda je príliš zdĺhavá, ale niekedy sa jej jednoducho nevyhneme. Preto vo väčšine prípadov používame druhú možnosť.

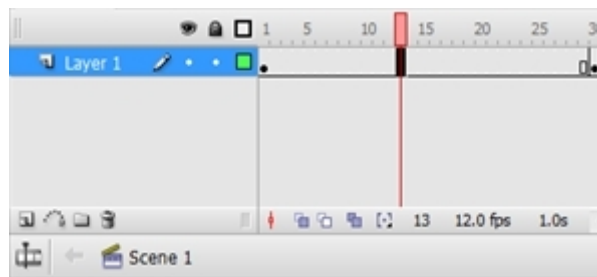
Animácia objektu pomocou Motion Tween

Pre názornú ukážku nám ako základný objekt postačí kruh, ktorý nakreslíme do prvej snímky na časovej osi (Time Line), pričom môžeme nastaviť efekt tieňa v *Color Mixer* (Shift+F9) na *Radial*.



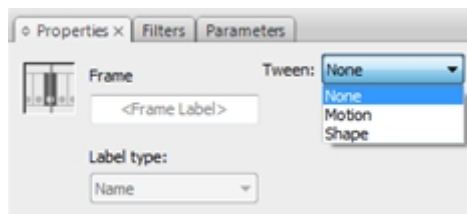
Obr.22 Farba výplne

Označíme kruh pomocou nástroja *Arrow Tool* (V) – čierna šípka a stlačíme F8 (Convert to Symbol). Z ponuky symbolov vyberieme *MovieClip*, zadáme názov a potvrdíme klávesom Enter. Ak máme otvorenú knižnicu (CTRL+L) zistíme, že po potvrdení objekt, ktorý sa stal symbolom, sa automaticky priradil do knižnice. Pre vytvorenie poslednej kľúčovej snímky na časovej osi sa nastavíme na snímku napr. číslo 30. Čím väčšie číslo zvolíme, tým dlhšie sa bude animácia prehrávať. Pri snímke číslo 30 bude dĺžka trvania animácie 2,4 sekundy, čo je pre nás plne postačujúce. Stlačíme F6, čím vytvoríme poslednú kľúčovú snímku (keyframe) animácie.



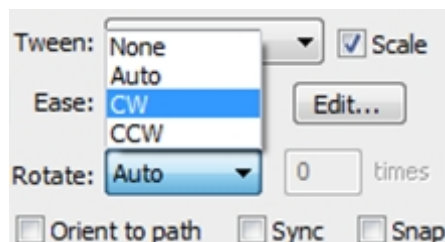
Obr.23 Vytvorenie kľúčových snímok animácie

Na poslednej snímke presunieme kruh na potrebné miesto. Keďže máme určenú prvú a poslednú snímku animácie, potrebujeme vykonať posledný krok – dopočítanie pohybu pomocou Motion Tweening. Nastavíme sa medzi kľúčové snímky a v paneli *Properties* v položke *Tween* potvrdíme *Motion*.



Obr.24 Vytvorenie pohybu pomocou Motion Tween

Animácia je hotová, stačí ju prehrať CTRL+Enter (Test Movie). Takto vytvorený kruh sa pohybuje, ale animácia nie je celkom reálna. Ideálne by bolo, keby sa pri pohybe kruh aj otáčal. Preto v poslednom kroku, kedy v paneli *Properties* nastavíme *Tween* na *Motion* musíme zároveň nastaviť aj smer rotácie nášho objektu (symbolu).



Obr.25 Nastavenie rotácie objektu

Ak zadáme v položke *Rotate* – *CW*, kruh sa bude otáčať vpravo, ak zadáme *CCW*, kruh sa bude otáčať vľavo.

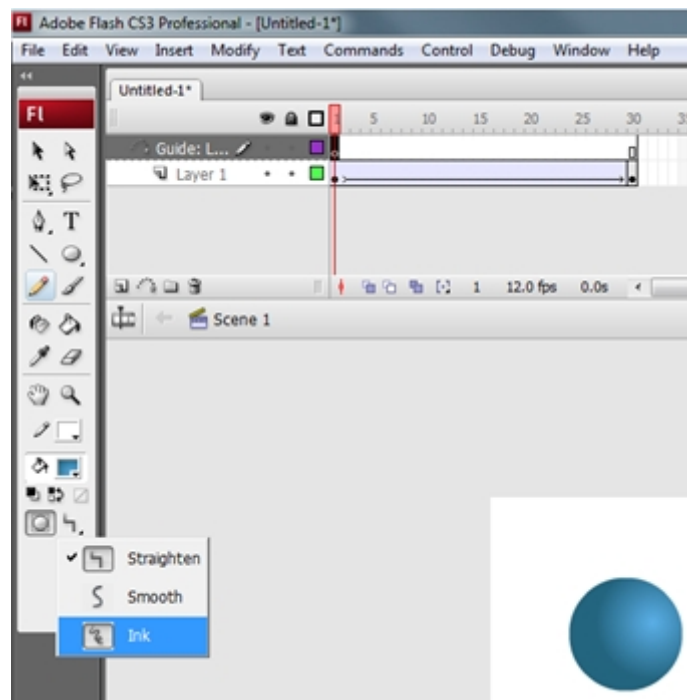
Pri modelovaní napr. fyzikálnych javov, sa často stretneme s problémom, kedy je potrebné pohyb v rámci animácie zrýchliť alebo naopak spomaliť. Typickým príkladom takéhoto pohybu je napríklad šikmý vrh. Na to môžeme použiť funkciu *Ease*, ktorá sa nachádza taktiež v paneli *Properties*. Rozsah je od -100 po 100. V prípade kladne zadanej hodnoty pôjde o postupné spomaľovanie, v prípade záporne zadanej hodnoty pôjde o postupné zrýchľovanie.

Takéto riešenie pohybujúceho sa kruhu je používané iba v najjednoduchších animáciách, kedy nie je potrebné zabezpečiť pohyb predmetu po nami zvolenej trase. Pri tvorbe multimediálnych aplikácií sa máme možnosť stretnúť i s pohybom, ktorý okrem toho, že je ovládaný tlačidlami alebo vstupnými parametrami, je aj pohybom po tzv. vodiacej čiare. Vodiaca čiara je čiara, po ktorej sa pohybuje náš objekt, pričom je možné meniť jeho parametre podobne ako v predchádzajúcom príklade – teda farbu, veľkosť, Alpha efekt atď..

Pre ukážku riešenia tohto druhu pohybu nám postačí predchádzajúci príklad pohybu. Na to, aby sa guľôčka pohybovala po zvolenej trase potrebujeme, vytvoriť jej vodiacu čiaru. Preto sa na časovej osi (Timeline) nastavíme na príslušnú vrstvu, kde sa nachádza naša guľôčka – v tomto prípade na Layer1. Príkazom *Insert* (v hlavnom menu) a *Motion Guide* vytvoríme vrstvu pre vodiacu čiaru. Do tejto vrstvy nástrojom ceruzka nakreslíme čiaru. Dôležité je, aby čiara nebola prerušená, pretože inak by sa pohyb nezobrazoval správne, resp. vôbec.

Spôsob kreslenia čiary máme možnosť zmeniť v *Tools* v položke *Options*. Na výber sú tri možnosti kreslenia:

1. pri kreslení sa čiara bude zakrivovať (zaostrovať) do hrán
2. pri kreslení sa čiara bude naopak snažiť hrany zaobľovať
3. pri kreslení čiara bude mať podobný efekt, ako keby sme ju nakreslili voľnou rukou.



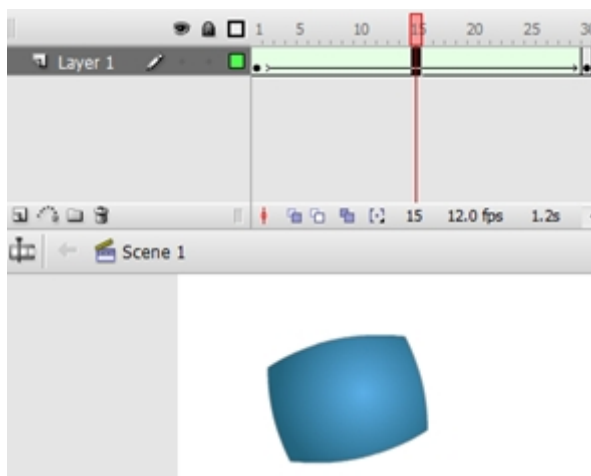
Obr.26 Vytvorenie vodiacej čiary

Po zvolení potrebného štýlu kreslenia ceruzky vytvoríme čiaru v snímke číslo 1, vo vodiacej vrstve *Motion Guide*. Pri vytvorení vodiacej vrstvy sa automaticky vytvorila aj jej dĺžka, ktorá obsahuje rovnaký počet snímok ako spodná vrstva. Ako posledný krok nám stačí nastaviť guľôčku v prvej snímke na začiatok vodiacej čiary a v poslednej snímke na koniec vodiacej čiary. Pri nastavovaní treba dbať nato, aby transformačný bod, ktorý je väčšinou umiestnený v strede *MovieClipu* bol umiestnený na vodiacej čiare. Pokiaľ zmeníme polohu transformačného bodu pomocou nástroja *Free Transform Tool* (Q) musíme zmeniť aj jeho polohu na vodiacej čiare.

Animácia objektu pomocou Shape Tween

Pri kreslení objektov sme si určite všimli, že sa skladajú z množstva malých bodov, tzv. pixlov aj napriek tomu, že program Adobe Flash pracuje na princípe vektorovej grafiky. Pokiaľ by sme chceli objekt prekonvertovať na *Symbol* a pri animácii sledovať zmenu jeho tvaru, zistíme, že to nie je možné. Doteraz sme totiž tvorili animácie tak, že sme použili funkciu *Motion Tween*, ktorá zmení iba veľkosť objektu, farbu, polohu, ale neumožní jeho deformáciu. V tomto prípade treba použiť funkciu *Shape Tween*. Ako príklad môžeme uviesť zmenu kruhu na obdĺžnik.

Pri deformácii tvaru objektu je veľmi dôležité, aby sme si uvedomili, že už s ním nemôžeme pracovať ako so symbolom typu *MovieClip*. Preto nakreslíme do snímky číslo 1 kruh. Na snímke číslo 30 vytvoríme poslednú kľúčovú snímku (F6), do ktorej namiesto kruhu nakreslíme obdĺžnik. Pokiaľ chceme, aby dochádzalo k deformácii objektu, musíme v paneli *Properties* nastaviť položku *Tween* na *Shape*. Efekt by sa mal prejavíť aj na časovej osi, kedy sa snímky označia bledozelenou farbou.

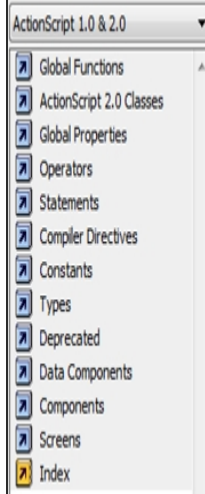


4.6 ActionScript v Adobe Flash

O tom, že Adobe Flash je skutočne silným univerzálnym grafickým nástrojom, ktorý sa oblasťou svojho pôsobenia nachádza medzi autorskými systémami a autorskými programovacími jazykmi, sme sa už zmienili. To, čo ho robí tak neuveriteľne príťažlivým, je najmä jednoduchosť skriptovania v spolupráci s možnosťami rastrových aj vektorových grafických programov. Predstavme si, že potrebujeme vytvoriť animované tlačidlo. Ak by sme ho mali naprogramovať v niektorom z bežných programovacích jazykov napr. Pascal, C alebo Basic či Java, určite by sme museli napísať najmenej 10 riadkov programu. A to nehovoríme o grafickom dizajne tlačidla, o možnosti použitia zvuku a podobne. V Adobe Flash je možné toto vykonať jednoduchým nakreslením tlačidla a až potom doplnením potrebného script-u, ktorý často pozostáva z minimálneho množstva riadkov (nie viac ako 5).

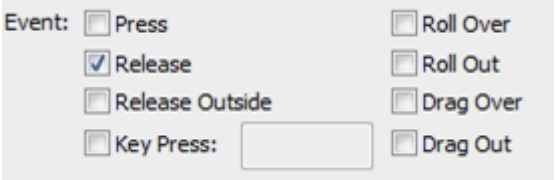
Základné príkazy

ActionScript má okrem svojho názvu veľmi veľa spoločného s JavaScriptom. Pri vysvetľovaní príkazov ActionScriptu sa obmedzíme iba na základné, najčastejšie používané príkazy. Panel Actions zobrazíme Windows/Actions (F9).

	Go To	prechod na snímku určenú číslom,
	Play	prehrávanie animácie,
	Stop	zastavenie prehrávania animácie,
	Stop All Sound	zastavenie prehrávania všetkých zvukov,
	GetURL	vkładanie dokumentov do MUME aplikácie, najčastejším príkladom je vloženie web stránky,
	FSCCommand	príkaz, ktorým je možné nastaviť chovanie MUME aplikácie v okne spustiteľného EXE súboru,
	Load Movie	nahrávanie externých SWF súborov bez nutnosti volať nový HTML dokument,
	UnLoad Movie	vymazanie nahratého externého SWF súboru,
	Tell Target	priradenie akcie objektu, alebo animácii v scéne na základe inštancie,
	If Frame Is Loaded	testuje stav snímky, ak je už nahratá, vykoná požadovanú akciu,
	On Mouse Event	priradenie udalostí a následný výkon akcie pri tlačidlách

Obr.28 Základné skupiny príkazov ActionScript 2.0

press	po stlačení tlačidla,
release	po stlačení a uvoľnení tlačidla,
Release Outside	po stlačení a opustení plochy tlačidla,

Key Press	po stlačení ľubovoľnej klávesy,
Roll Over	pokiaľ sa nachádzame nad aktívnou plochou tlačidla,
Roll Out	po odchode z aktívnej plochy tlačidla,
Drag Over	po stlačení tlačidla a jeho následnom premiestnení von a naspäť naň,
Drag Out	po stlačení tlačidla a jeho následnom premiestnení von z neho
	


Obr.29 Možnosti akcií príkazu On Mouse Event pri aktivovanom Script Assist

Práca so vstupným a výstupným poľom

Pri formulároch, ktoré nám môžu dobre poslúžiť aj napr. pri testovaní vedomostí používateľa MUME aplikácií ide väčšinou len o jedno. Načítať text zo vstupného poľa a poslať ho k ďalšiemu spracovaniu. Na zjednodušenom príklade si môžeme ukázať načítanie textu, overenie jeho správnosti a zobrazenie počtu nesprávnych pokusov.

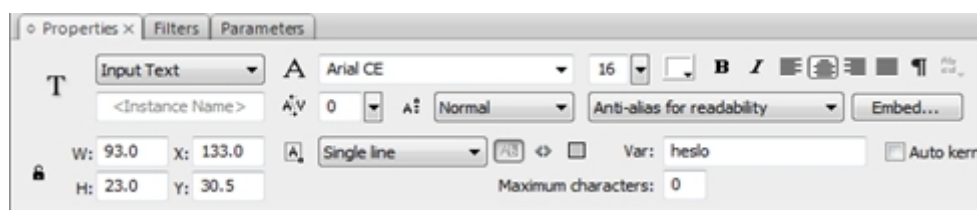
Zadajte heslo:

Počet neúspešných pokusov:



Obr.30 Príklad práce so vstupným a výstupným poľom

Po otvorení nového dokumentu v Adobe Flash si vytvoríme vstupné textové pole podobne, ako keď vkladáme text. V paneli Properties však nastavíme vlastnosť textového poľa na Input Text. Tým zabezpečíme, že môžeme vkladať potrebné znaky, ktorých správnosť sa bude neskôr testovať. Vstupné textové pole je potrebné pomenovať ako variabilnú premennú – v našom prípade „heslo“.

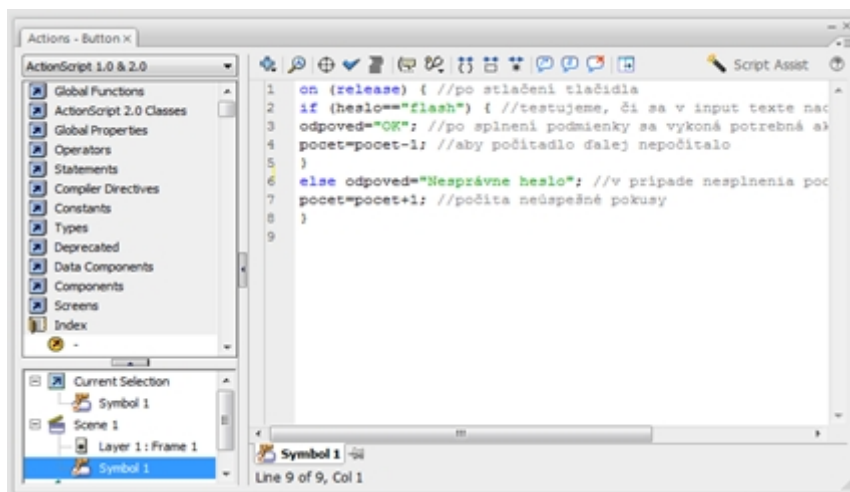


Obr.31 Definovanie premennej pre vstupné pole

Na vytvorenie takéhoto druhu testovania potrebujeme tlačidlo, v ktorom sa bude nachádzať script akcie pre zisťovanie správnosti zadaného výrazu vo vstupnom poli. Pre informatívny výpis napr. počtu neúspešných pokusov, ako aj výpisu, či je výraz vo vstupnom poli správny alebo nesprávny je potrebné vložiť výstupné pole, ktoré vytvoríme podobne ako vstupné, ibaže jeho vlastnosť zmeníme v paneli Properties na Dynamic Text. Taktiež je potrebné tieto dynamické polia pomenovať ako určitú premennú.

Pri práci s textovým poľom si musíme dávať pozor, pretože ak chceme opäť použiť funkciu textového poľa, Adobe Flash si pamätá posledný spôsob použitia. Preto sa môže niekedy stať, že budeme pracovať napr. s výstupným poľom, aj keď sme chceli vložiť statický text. O tom, aký typ textového poľa práve používame, je možné sa

presvedčiť v paneli Properties. Pre vloženie príkazov tlačidla určeného na testovanie je potrebné zobrazíť panel vkladania akcií (Actions, F9).



Obr.32 Príklad ActionScriptu tlačidla „Stlať“

Pozor! Je veľmi dôležité, aby sme príkazy vkladali do správneho objektu, v tomto prípade tlačidla. O tom, kde príkazy jazyka ActionScript vkladáme, nás informuje dolný ľavý roh panela Action. Potrebne príkazy zadávame do tlačidla pomocou jednoduchého menu.



Obr.33 Menu ActionScriptu

Popis tlačidiel menu ActionScript (zľava do prava):

- vkladanie príkazov jazyka ActionScript, môžeme ich vkladať aj cez položku Index
- vyhľadanie výrazu
- priradenie akcie konkrétnemu objektu alebo animácií na scéne
- kontrola syntaxe príkazov
- zarovnanie príkazov ActionScriptu podľa štandardu
- výber z možností príkazov
- zvyšné možnosti Action menu sa používajú iba zriedkavo.

Príkazy pre testovanie správnosti zadaného výrazu vo vstupnom poli:

```
if (heslo=="flash") { //testujeme, či sa v Input Text nachádza daný výraz
```

```
    odpoved="OK"; //po splnení podmienky sa vykoná potrebná akcia
```

```
    pocet=pocet-1; //a počítadlo nebude ďalej počítat'
```

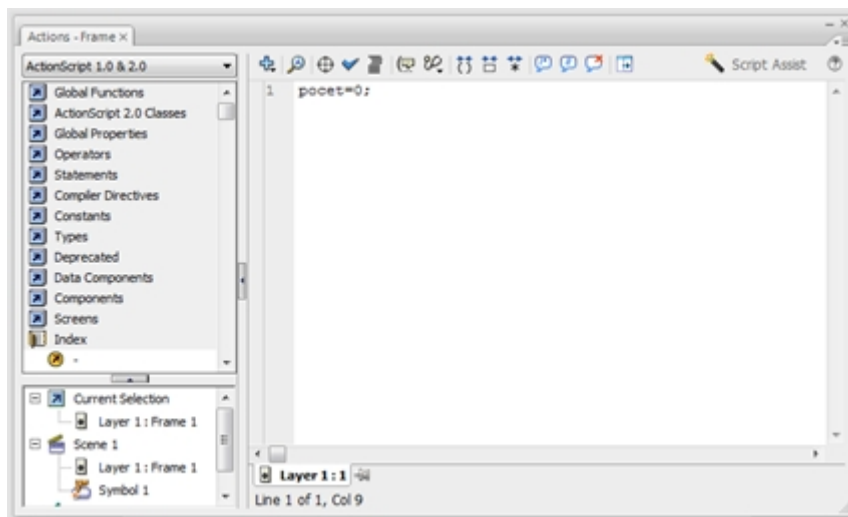
```
}
```

```
else odpoved="Nesprávne heslo"; //v prípade nesplnenia podmienky sa vykoná iná akcia
```

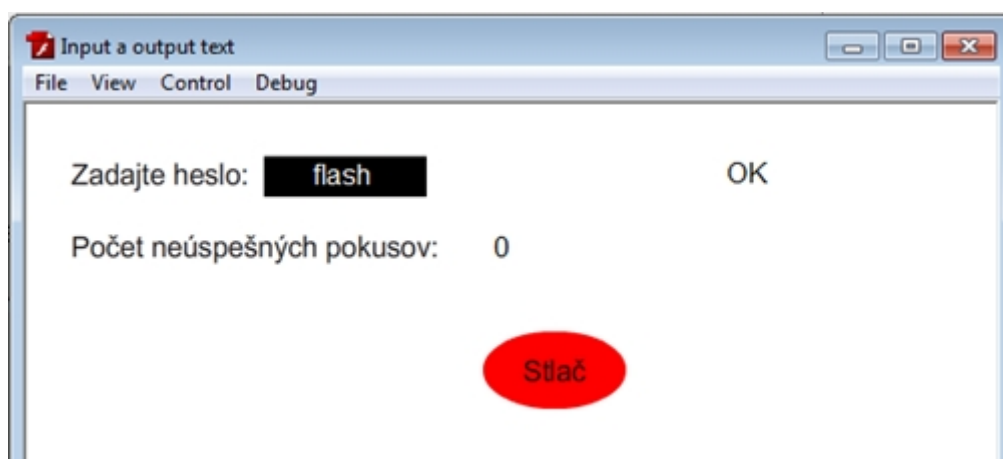
```
pocet=pocet+1; //a počítajú sa neúspešné pokusy
```

```
}
```

Po spustení a otestovaní (CTRL+Enter) však zistíme, že počítadlo nepracuje úplne správne, ale priraduje vedľa seba číslo 1 (ak sme nastavili počítanie od 1). Na odstránenie tohto nedostatku je potrebné počítadlo nastaviť na hodnotu 0. Je to preto, aby sme určili, od akého čísla má začať počítat'.



Obr.34 Definovanie začiatkovej hodnoty premennej



Obr.35 Ukážka formuláru vytvoreného vstupným a výstupným poľom

Načítanie textu z externého zdroja (tvorba scrollbaru)

V predchádzajúcej kapitole sme si ukázali prácu so vstupným a výstupným poľom. Dynamický text je možné použiť nielen ako počítadlo prístupov, či informatívny výpis, ale aj ako jednoduchý spôsob načítania textu z externého zdroja. V prípade, že potrebujeme vykonať aktualizáciu údajov, je to určite pohodlnejší spôsob, ako upravovať súbor SWF. Ako zdroj dát použijeme textový dokument vytvorený v poznámkovom bloku (Notepad). Na začiatok textu napíšeme meno textového poľa, do ktorého budeme vkladat potrebný naformátovaný text, a oddelíme ho operátorom „=”.

Napríklad: pole = Toto je text, ktorý chceme zobrazit' pomocou vytvoreného scrollbaru v Adobe Flash.

Takto vytvorený text by sa však v textovom poli zobrazil bez formátovania. Niekedy je ale potrebné, aby sme zvýraznili napr. nadpisy, hypertextové prepojenia, zmenili farbu textu, alebo zalomili riadok odseku. To všetko je možné vykonať pomocou formátovacích príkazov jazyka HTML.

Ako príklad uvedieme formátovanie textu zmenou rezu a farby písma:

pole=<html><body><p align="justify"> Toto je text, ktorý chceme zobrazit' pomocou vytvoreného scrollbaru v Adobe Flash.

Toto je text, ktorý chceme zobrazit' pomocou vytvoreného scrollbaru v Adobe Flash.

<i>Toto je text, ktorý chceme zobrazit' pomocou vytvoreného scrollbaru v Adobe Flash.</i>

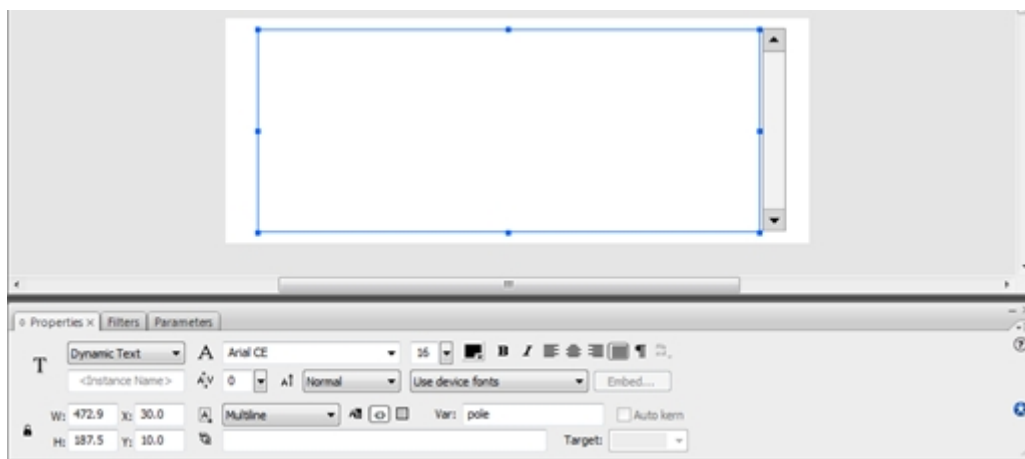
<u>Toto je text, ktorý chceme zobrazit' pomocou vytvoreného scrollbaru v Adobe Flash.</u>


```

<b><i>Toto je text, ktorý chceme zobrazit' pomocou vytvoreného scrollbaru v Adobe Flash.</i></b>
<br>
<p><font color="#FF0000">Toto je text, ktorý chceme zobrazit' pomocou vytvoreného scrollbaru v
Adobe Flash.<p><br>
<b><u>Toto je text, ktorý chceme zobrazit' pomocou vytvoreného scrollbaru v Adobe Flash.</u><b>
<br></p></body></html>

```

Textový súbor je potrebné uložiť v kódovaní UTF-8, aby sa správne zobrazovala diakritika v textovom poli. Pokiaľ už máme vytvorený textový dokument, ktorého obsah budeme načítavať, môžeme pristúpiť k samotnej tvorbe Scrollbaru. Jeho základom bude dynamický text, ktorému nastavíme potrebné parametre, ako *Multiline* (aby sa text automaticky zalamoval), *Render text as HTML* (aby sme správne zobrazili formátovanie) a nakoniec priradiť mu funkciu variabilnej premennej, v našom prípade pole.



Obr.36 Definovanie dynamického poľa pre text

Do snímky číslo 1 vložíme nasledovný kód: `loadVariablesNum("text.txt", 0);`

Tým zabezpečíme, že do pôvodne prázdneho poľa sa načíta text, ktorý sme vytvorili v súbore pod názvom text.txt. Adobe Flash má oproti klasickým HTML stránkam jednu nevýhodu. Neumožňuje totiž žiadnou zo štandardných komponent, ktoré sa v ňom nachádzajú, akýmkoľvek spôsobom posúvať text. Na odstránenie tohto problému však môžeme vytvoriť jednoduché tlačidlá, pomocou ktorých budeme posúvať text hore alebo dole.

Do tlačidla, ktoré bude predstavovať posun textu smerom hore, vložíme nasledovný kód:

```

on (press) {
    hodnota = -1;
}
on (release) {
    hodnota = 0;
}

```

Do tlačidla, ktoré bude predstavovať posun textu smerom dole, vložíme nasledovný kód:

```

on (press) {
    hodnota = 1;
}

```

```

}

on (release) {

hodnota = 0;

}

```

Takto vytvorené tlačidlá s priradenými akciami označíme a vytvoríme z nich nový symbol – *MovieClip* a priradíme mu nasledujúcu akciu:

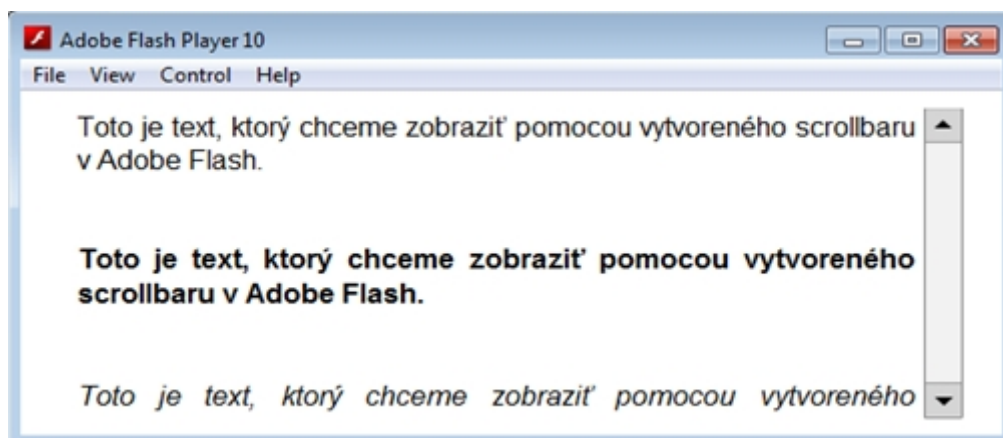
```

onClipEvent (enterFrame) {

_root.pole.scroll = _root.pole.scroll+this.hodnota;

}

```



Obr.37 Ukážka načítania textu do dynamického poľa

Vyššie uvedený script nám zaistí posun textu v oboch smeroch vďaka vlastnosti *scroll* pri zmene hodnoty premennej *pole*. Pred samotným otestovaním vytvoreného scrollbaru je potrebné súbor Adobe Flash uložiť. Preto ho ukladáme tam, kde je uložený aj textový súbor, z ktorého sa načítavajú údaje. Otestovanie, či sa správne načítavajú potrebné údaje, vykonáme stlačením *CTRL+Enter*.

Načítanie obrázkov a animácií pomocou funkcie *loadMovie*

Pre načítanie externých SWF alebo JPEG súborov, ktoré môžeme použiť napr. ako pozadie a tak zmenšiť celkovú veľkosť súboru, slúži funkcia *loadMovie*.

```
loadMovie("url", názov);
```

url – Predstavuje absolútnu alebo relatívnu adresu umiestnenia súboru swf alebo JPEG, ktorý sa má načítať. Najčastejšie sa používa relatívna adresa.

názov – Predstavuje názov *MovieClipu* v Adobe Flash, do ktorého chceme načítať animáciu.