



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ



UNIVERSITAS
OSTRAVIENSIS

TVORBA ANIMACÍ

(metodická příručka pro autory)

Martin Němec

Ostrava 2011

Název: Tvorba animací
Autor: Martin Němec
Vydání: první, 2011
Počet stran: 30

Jazyková korektura: nebyla provedena.

Určeno pro projekt:

Operační program Vzdělávání pro konkurenceschopnost

Název: Adaptivní individualizovaná výuka v e-learningu

Číslo: CZ.1.07/2.3.00/07.0019

Realizace OU – Ostravská univerzita v Ostravě

Projekt je spolufinancován z prostředků ESF a státního rozpočtu ČR

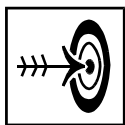
© Martin Němec

© VŠB – Technická univerzita Ostrava

Obsah

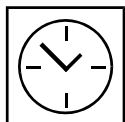
1	Úvod.....	4
1.1	Vývoj programu Flash.....	4
2	Základní prostředí Adobe Flash.....	6
2.1	Popis vývojového prostředí.....	7
2.2	Možnosti editace.....	10
2.3	Barvy a jejich míchání.....	11
2.4	Zarovnávání objektů a jejich transformace.....	12
2.5	Základní objekty.....	14
2.6	Časová osa.....	14
3	Základní metody doplňování.....	16
3.1	Doplnění pohybu.....	16
3.2	Doplnění tvaru.....	17
3.3	Klasické doplnění.....	18
3.4	Maska (animace maskou).....	19
3.5	Symbol Tlačítko (Button).....	20
4	Vytváření aplikace.....	21
4.1	Vyšší využití Adobe Flash.....	21
4.2	Ukázka vytváření ActionScriptu a porovnání verzí.....	22
4.3	Publikování výsledného projektu.....	23
4.4	Online nápověda.....	24
5	Závěr.....	25
	Příloha základních praktických příkladu.....	26
	Použitá literatura.....	31

1 Úvod



Cíl kapitoly

Tato kapitola seznamuje uživatele se základním cílem této příručky a vlastnostmi a vývojem aplikace Adobe Flash CS 5.5.



Čas na prostudování kapitoly

Čas pro nastudování této kapitoly je asi 30 minut.

Cílem tohoto materiálu je přinést základní přehled a možnosti s návodem použití zejména pro začínající uživatele programu Adobe Flash. Aktuální verze Adobe Flash CS5.5 patří mezi velmi silné nástroje pro tvorbu interaktivního obsahu multimediálních aplikací, který lze použít jak pro samotné „standalone“ aplikace (dodané např. na DVD) nebo jako webovské prezentace. Již od počátku bylo jeho hlavní výhodou použití vektorové grafiky a postupně s rozšiřováním ActionScriptu možnost celkem jednoduše vytvářet i rozsáhlejší interaktivní aplikace. S novou verzí Adobe Flash CS5.5 přichází také podpora rozšíření na další platformy, mezi které patří Android, Apple iOS a Adobe AIR.

1.1 Vývoj programu Flash

Na začátku byl program Flash nástupcem animovaného GIFu (formát GIF 89a, rok 1989), který umožňoval postupné přehazování obrázku Animovaný GIF, čímž bylo docíleno efektu videa, bez možnosti zásahu do posloupnosti snímků s omezenou paletou barev. S příchodem programu Flash, v té době vlastněný firmou Macromedia, vznikla možnost vytvářet interaktivní animace, aplikace, prezentace a hry s malou výslednou velikostí, k čemuž dopomohla možnost použití vektorového formátu. Následně Flash postupně vytlačoval doté doby používaný GIF a získával na oblibě.

Během postupného vývoje programu Flash až k dnešní verzi označené CS5.5, prošel velkým vývojem včetně vnitřního programovacího jazyka ActionScript. Zde postupně od skriptovacího jazyka došla až po plně objektově orientovaný jazyk s označením ActionScript 3.0, umožňující velkou škálu rozšiřujících možností v oblasti interaktivity.

V dnešní době již existují konkurenční nástroje lišící se v některých vlastnostech a možnostech. Jako příklad uveďme platformu Silverlight vytvářenou společností Microsoft, která patří mezi hlavní konkurenty. Dalším konkurenčním nástrojem by měla být rozšiřující specifikace jazyka HTML s názvem HTML5, vyvíjená společností W3C.

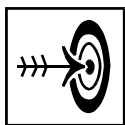
Mezi základní výhody použití programu Flash patří:

- dostupnost
- bezpečnost
- jednoduchost
- vektorové animace
- velikost

Tvorba animací

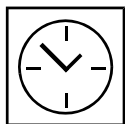
- práce s videem, zvukem
- interaktivita
- použití lokálně i na internetu

2 Základní prostředí Adobe Flash



Cíl kapitoly

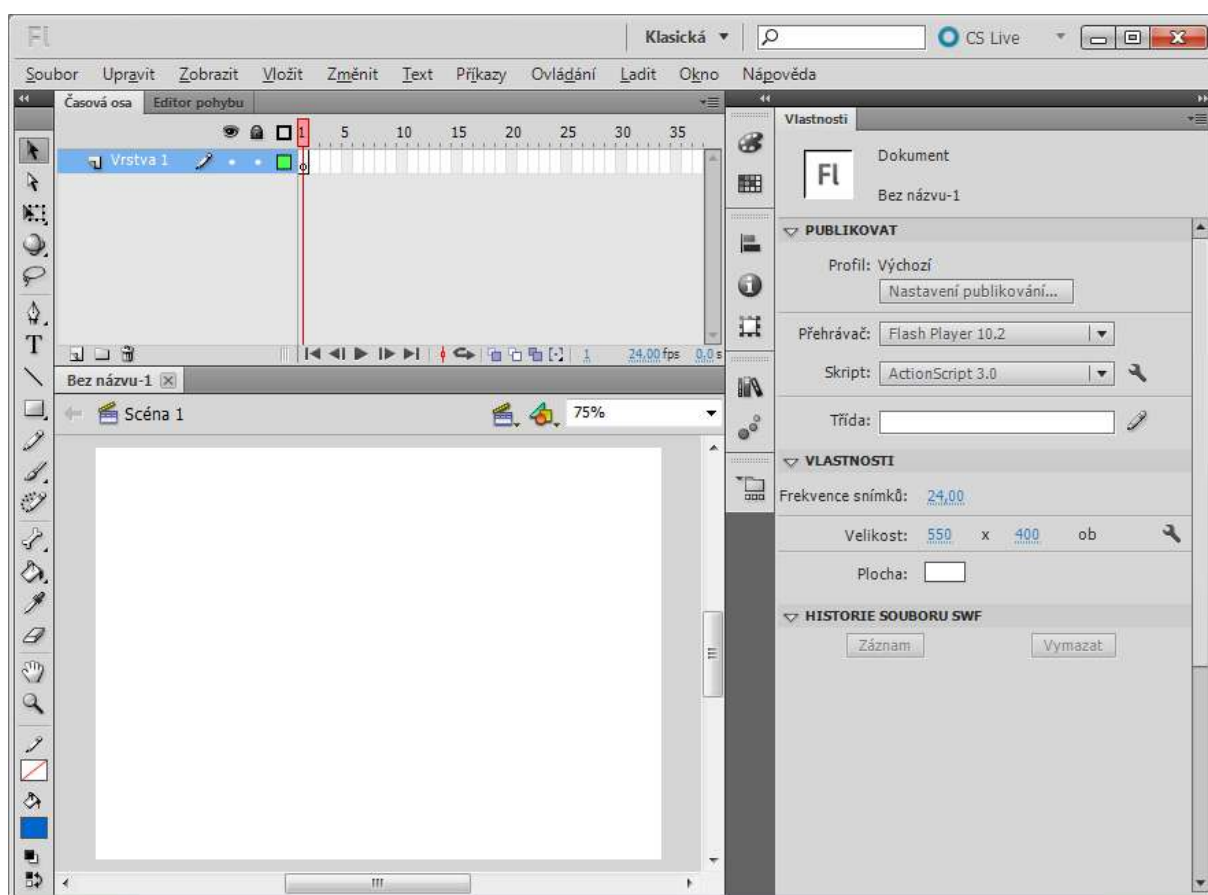
Tato kapitola se postupně zabývá popisem vývojového prostředí programu Adobe Flash CS5.5, možnostmi editace těles, jejich transformací a použitím barev. Dále popisuje tři základní symboly a použití časové osy.



Čas na prostudování kapitoly

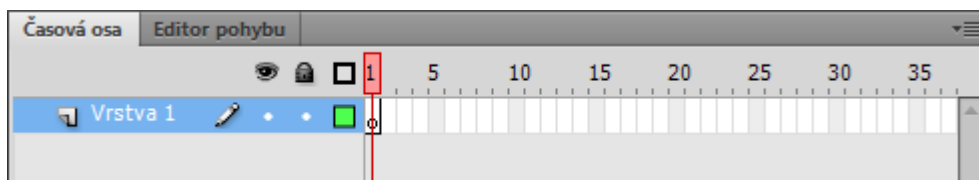
Čas pro nastudování této kapitoly je asi 200 minut.

Na Obrázku 1. je zobrazeno hlavní prostředí komerčního nástroje Adobe Flash, který však lze získat i jako 30 denní zkušební verzi přímo ze stránek [1].



Obrázek 1: Vývojové prostředí Adobe Flash

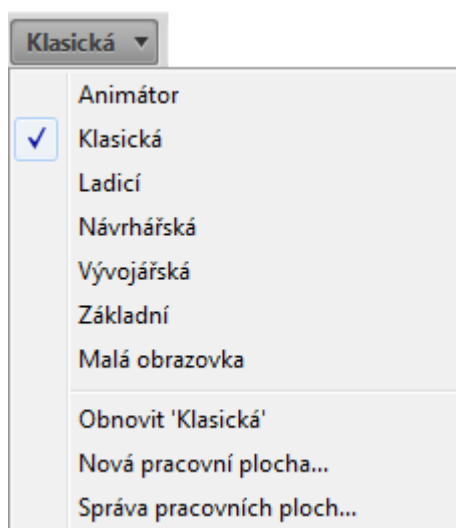
Vývoj aplikací v programu Adobe Flash je obvykle vztažen k časové ose (obr. 2), která v podstatě obsahuje jednotlivé snímky výsledné animace. Animace je pak závislá na nastavení rychlosti přehrávání snímku za vteřinu. Vytvořit animaci o délce pěti vteřin bychom mohli v tomto případě vytvořením jednotlivých snímků na časové ose, kterých by při nastavení 24 snímků za vteřinu celkem 120 snímků.



Obrázek 2: časová osa

Vytvářet všechny snímky ručně by samozřejmě bylo hodně pracné a lze využít časovou osu a její možnosti sestavovat animace pomocí různých efektů (posun, rotace, změna velikosti, tvarů, barvy, textu apod.), které se nemusí plně tvořit ručně, ale využít nástroje programu Adobe Flash.

Při vývoji aplikace je vhodné si nejdříve upravit vlastní grafické uživatelské rozhraní, k dispozici je několik předdefinovaných variant (obrázek 3.), které však umožňují uživateli upravovat jejich výsledný vzhled. Mimo to může uživatel nadefinovaný vzhled uložit jako novou pracovní plochu a kdykoli se k tomuto rozvržení vrátit. Vlastní rozvržení je závislé nejen na zvyklostech uživatele, ale také na rozlišení použitého monitoru apod.



Obrázek 3: Volba vzhledu GUI

Samotné prostředí Adobe Flashe lze rozdělit do několika panelů, které lze různě zapínat a upravovat. Mezi základní části patří panel nástrojů, časová osa, pracovní plocha, nastavení.

Jednoduchý postup při tvorbě základní Adobe Flash aplikace lze rozdělit do několika částí. Mezi tyto kroky patří:

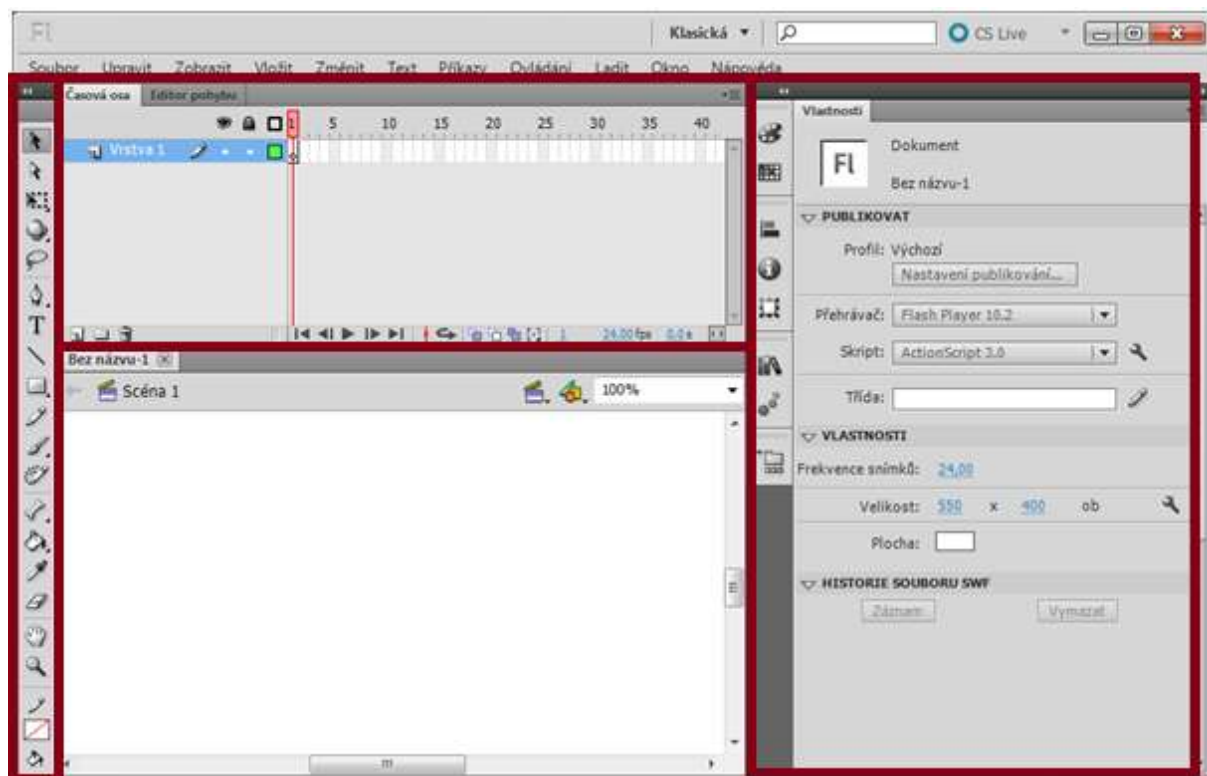
- Vytvoření základních objektů používaných na snímcích.
- Vytvoření jednotlivých animací.
- Přidání interaktivity (doplnění o skripty).
- Publikování výsledného projektu.

2.1 Popis vývojového prostředí

Mezi nejdůležitější panely programu Adobe Flash patří tyto panely (obr. 4.):

- **Panel Nástrojů** (CTRL+F2) – obsahuje základní ikony pro vytváření jednotlivých snímků
- **Panel Vlastností** (CTRL+F3) – umožňuje nastavovat základní vlastnosti

- **Panel Časové osy (CTRL+ALT+T)** – zobrazuje panel s časovou osou a jednotlivými vrstvami



Obrázek 4: Pracovní plocha s panely (zleva: Nástroje, Časová osa, Vlastnosti)

Pro základní tvorbu obsahu snímku je nejdůležitější seznámit se s panelem nástrojů, ten se přizpůsobuje velikosti, kterou je možné měnit najetím kurzoru myši na okraj panelu, a stisknutím tlačítka můžeme navolit libovolnou šířku panelu. Ikony uvnitř panelu se přizpůsobují velikosti okna. Na obrázku 5. je šířka upravena tak, aby se všechny ikony zobrazily v jednom řádku.



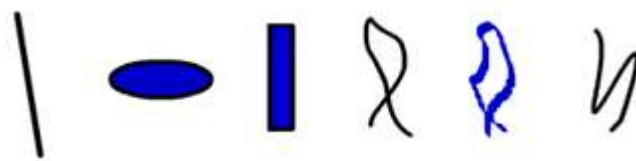
Obrázek 5: Zobrazení panelu nástroje

Panel nástrojů umožňuje následující možnosti:

- **Nástroj výběru (V)** - posunování s objekty, editace již vytvořených objektů, přesouvání
- **Nástroj dílčí výběr (A)** - editace již vytvořených čar, objektů a křivek
- **Nástroj volná transformace (Q)** - změna polohy a vzhledu objektů (přechodu)
 - **Rotare and Skew** - změnu rotace, ale také můžeme změnit zkosení objektu
 - **Scale** - změna velikosti zvoleného objektu
 - **Distort** - změnit tvar rohů u zvoleného objektu
 - **Envelope** - pomocné body umožňující dokonalejší editaci tvaru

- **Nástroj 3D natočení (W)** - umožňuje 3D rotace s objekty (natočení a posunutí)
- **Nástroj Laso (L)** - slouží k přesnějšímu výběru objektů i jejich částí
- **Nástroj pero (P)** - slouží k vytváření objektů pomocí klíčových bodů
- **Textový nástroj (T)** - vkládání textů a jejich editace
- **Nástroj čára (N)** – vytváření úseček (SHIFT - rovné úsečky)
- **Nástroj obdélník (R)** - kreslení čtverců a obdélníků zvolené barvy a výplně (změnu barev můžete upravit v liště Colors). Ve spodní části panelu (Options) máte možnost změnit zaoblení hran. Součástí je i volba **Nástroj elipsa (O)** - vytváření kruhů a elips.
- **Nástroj tužka (Y)** - kreslení úseček od ruky. Straighten (ostré zarovnání), Smooth (oblé zarovnání), Ink (neupravuje nic)
- **Nástroj štětec (B)** - kreslení křivek, které připomínají tah štětce.
- **Nástroj deco (U)** - dekorativní motivy
- **Nástroj kost (X)** - výběr objektů
- **Nástroj plechovka barvy (K)** - vyplnění objektu danou barvou či gradientem. Součástí je i **Nástroj kalamář (S)** - barva ohraničení
- **Nástroj kapátko (I)** - zjištění barvy daného objektu. Následně po vybrání se aktivuje Nástroj plechovka barvy
- **Nástroj guma (E)** - mazání jednotlivých objektů či částí, které jsou umístěny ve scéně klipu
- **Nástroj ručička (Q)** - přesunování celého dokumentu
- **Nástroj lupa (M)** - zvětšování a zmenšování dokumentu v procentech

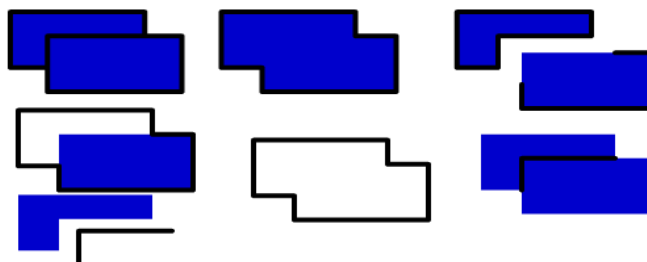
Mezi základní objekty patří tyto objekty *Nástroj čára (N)*, *Nástroj elipsa (O)*, *Nástroj obdélník (R)*, *Nástroj tužka (Y)*, *Nástroj štětec (B)*, *Nástroj pero (P)*. Příklad použití těchto nástrojů jsou vidět na obrázku 6.



Obrázek 6: Základní objekty pro kreslení

Jednotlivé objekty se obvykle skládají z hrany a výplně. Obě částí lze vybírat a nastavovat jejich vlastnosti. Mezi základní vlastnosti samozřejmě patří barva, tloušťka apod.

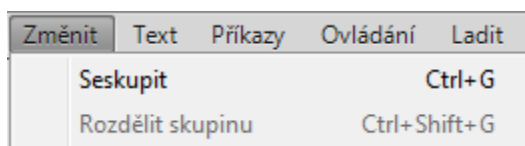
Složitější objekty samozřejmě nelze vytvářet pouze výše popsanými základními objekty, ale využívá se jejich kombinace, která se vytvoří například překrytím jednotlivých částí základních objektů. Na obrázku 7. lze vidět několik příkladu, jak lze kombinací primitivních útvaru získávat složitější tělesa.



Obrázek 7: Kombinace několika těles

Jednotlivé objekty a jejich části, lze také slučovat do skupin nebo zpětně rozkládat (obrázek 8.). Výhodou je to, že se sloučené objekty ve výsledku chovají jako jeden objekt, ale lze jej kdykoli rozložit a upravovat jednotlivé vnitřní části. Pro slučování vybraných částí, respektive vytváření a zrušení skupiny se používají následující příkazy:

- **Seskupit** (Ctrl+G) - vytvoření skupiny z vybraných objektů
- **Rozdělit skupinu** (Ctrl+Shift+G) - zrušení skupiny

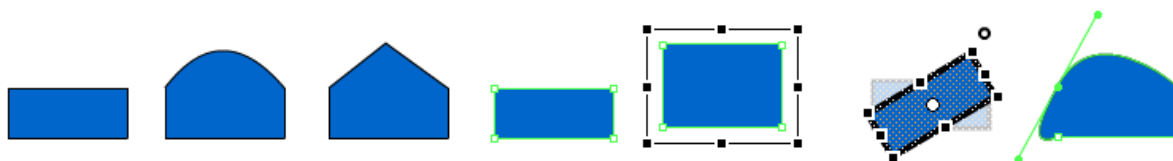


Obrázek 8: Nabídka seskupit a rozdělit skupinu

2.2 Možnosti editace

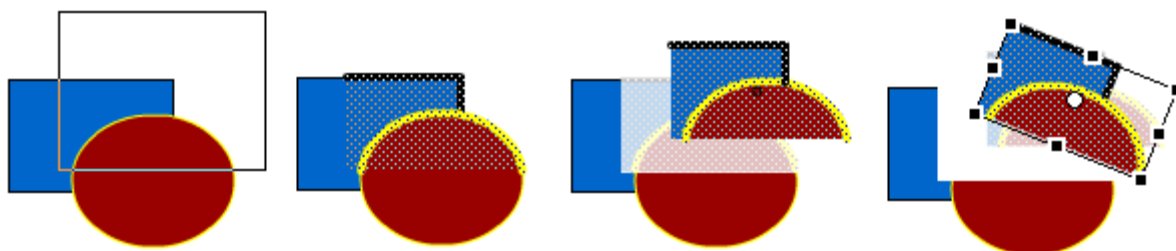
Jak již bylo výše řečeno, lze každý objekt rozdělit na obsah a hranici. Samozřejmě jde následně upravovat celý objekt i obě jednotlivé části (obrázek 9.). Mezi základní možnosti editace patří úprava okraje tělesa.

- Možnost vytváření křivky z hrany, s přidruženým ALT vytvářet další vrcholy na místě kliknutí
- Možnost editovat hrany a křivky
- Možnost nastavovat velikost, zkosení, rotaci (kolečko střed)



Obrázek 9: Základní transformace provedené na tělese obdélník

Upravovat lze jak celé těleso tak jeho jednotlivé části (jednotlivé hrany nebo výplň tělesa). Samozřejmě lze upravovat také pouze výběrem označenou oblast, která může být tvořena jedním nebo více objekty (obrázek 10.). Výběr oblasti lze učinit kliknutím myši na určitou část a výběrovým obdélníkem potřebnou část vybrat. Útvar, který je složen z jednotlivých vybraných částí se následně chová jako jediné těleso.

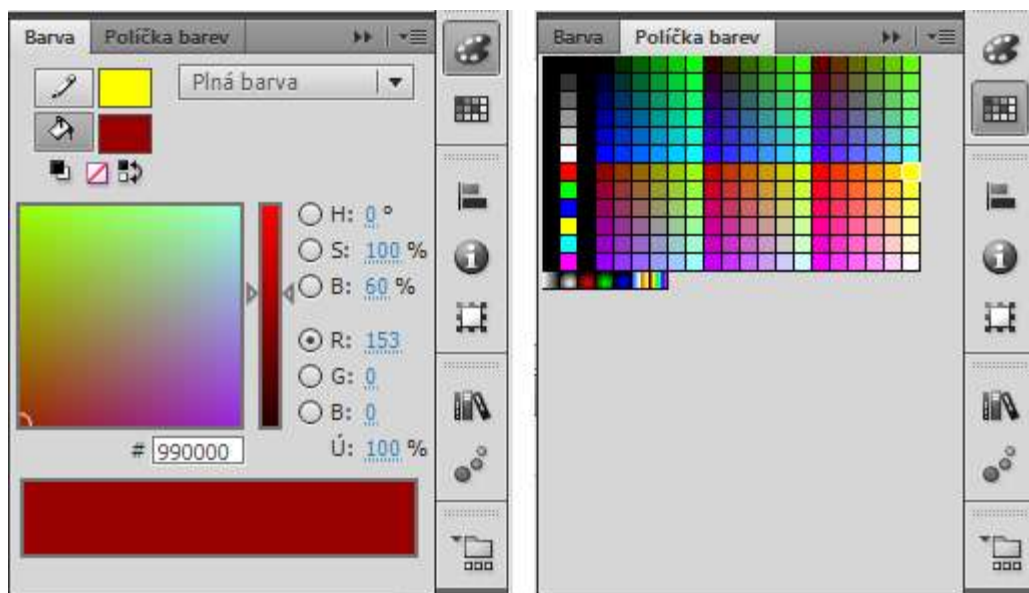


Obrázek 10: Modifikace vybrané oblasti

Výběr několika částí nebo vybraných oblastí lze uskutečnit pomocí stisknutí klávesy SHIFT.

2.3 Barvy a jejich míchání

Barva patří mezi základní vlastnosti každého objektu, ten lze rozdělit na barvu okraje (barva tahu) a barvu obsahu. Pro výběr barvy lze použít panel *Barva* nebo panel *Políčka barev*.



Obrázek 11: Panely určené pro míchání barev (panel Barva, panel Políčka barev)

Panel Barva je určen k míchání libovolného typu barvy, naproti tomu panel Políčka barev se skládá z tabulky použitých barev a umožňuje tak rychleji vybrat již použitou barvu.

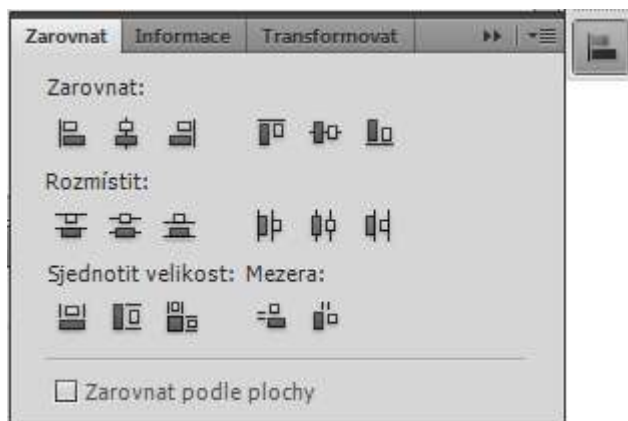
Panel Barva umožňuje míchání výplně tělesa několika možnostmi

- **Neurčeno** - žádná výplň.
- **Plná barva** - základní výplň celého objektu
- **Lineární přechod** - vyplní objekt lineárně dvěma (nebo i více) barvami.
- **Radiální přechod** - výplň odpovídá tvaru kruhu.
- **Bitmapová výplň** - použití rastrového obrázku (textura)

Namíchaná barva se navíc zobrazuje jako číselná hodnota v šestnáctkovém zápisu (Hexa kódu), v barevném modelu HSB (postupně nastavujeme barevný tón, sytost a jas) nebo v barevném modelu RGB (červená, modrá a zelená barva) včetně alfa kanálu.

2.4 Zarovnávání objektů a jejich transformace

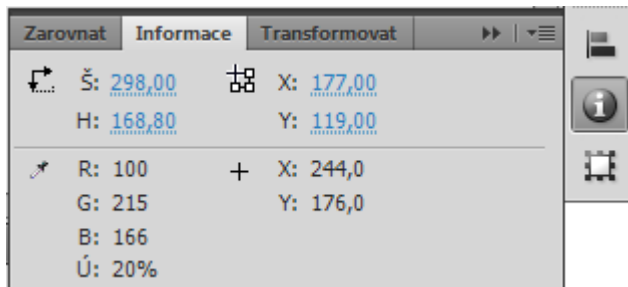
Jednou z dalších velmi důležitých možností Adobe Flash je zarovnávání několika vybraných objektů a jejich transformace. Pro vyvolání těchto možností slouží nabídka Zarovnat. První záložka této nabídky (obrázek 12.) umožňuje zarovnávat dva a více vybraných objektů vodorovně nebo svisle a to na střed, levý (horní) nebo pravý (dolní) okraj.



Obrázek 12: Záložka Zarovnat na panelu Zarovnat

Další možností je rozmístění, které umožňuje rozmístit objekty podle středů nebo jednotlivých okrajů.

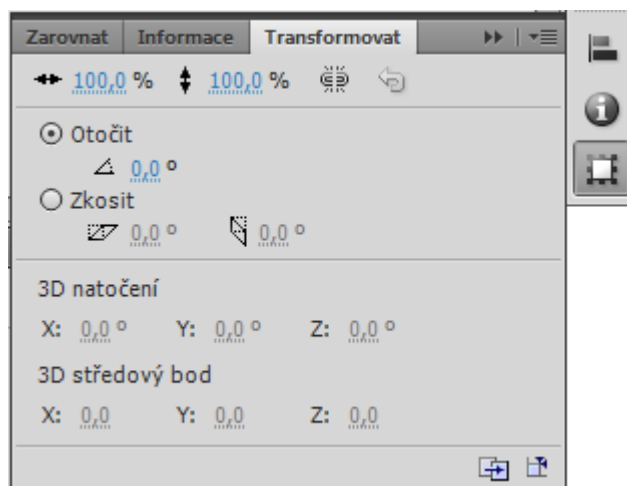
Poslední volbou této záložky je Sjednocení velikosti (šířka, výška nebo obojí) a nastavení mezer ve svislém nebo vodorovném směru.



Obrázek 13: Panel Informace

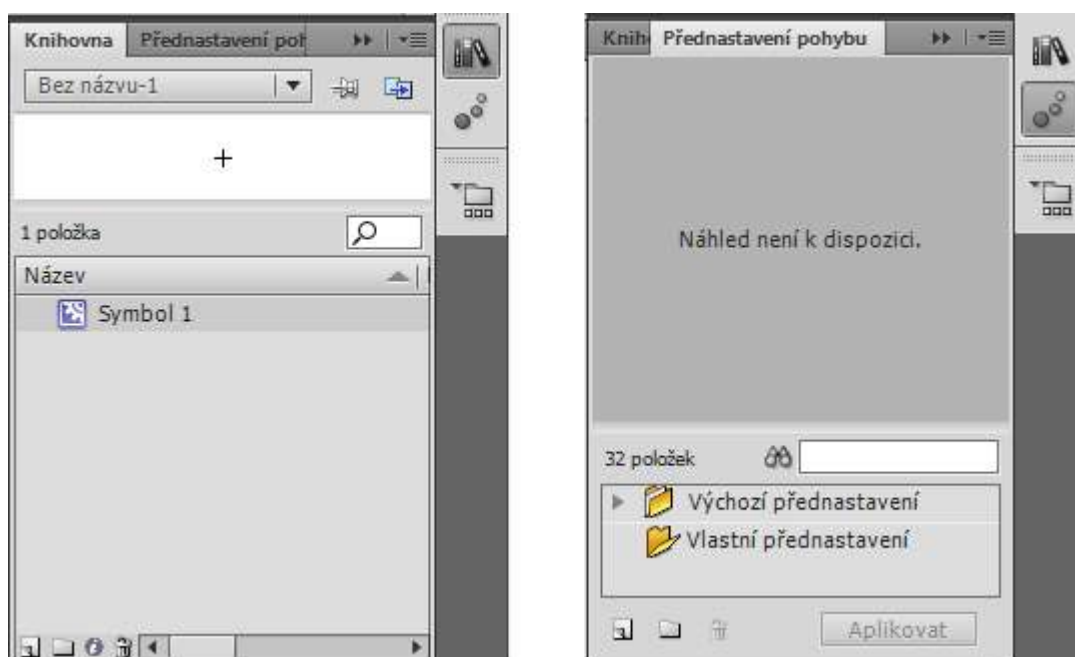
Panel Informace (obrázek 13.), zobrazuje základní informace o šířce a výšce výběru, poloze transformačního bodu, barva a poloha kurzoru.

Dalším panelem je Transformace (obrázek 14.), která umožňuje přesně definovat základní transformace, mezi které patří šířka a výška měřítka, rotace (otočení), zkosení a nastavení 3D rotace včetně středového bodu.



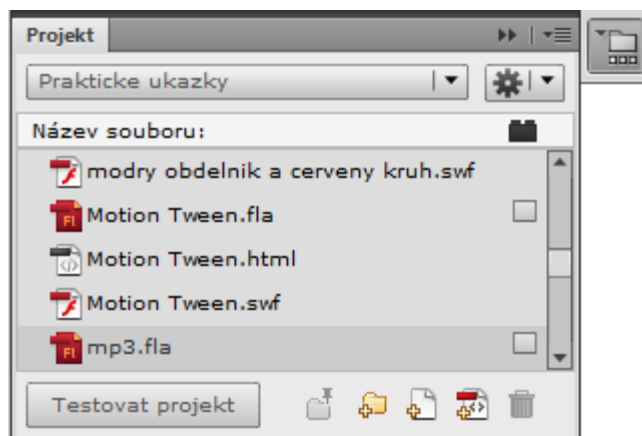
Obrázek 14: Panel Transformace

Dalšími dvěma panely jsou Knihovna a Přednastavení pohybu (obrázek 15.). Panel knihovna obsahuje jednotlivé objekty, použité ve scéně s možností náhledu. Panel přednastavení pohybu obsahuje předem vytvořené pohyby, které lze jednoduše přiřazovat jednotlivým objektům.



Obrázek 15: Panely Knihovna a Přednastavení pohybu

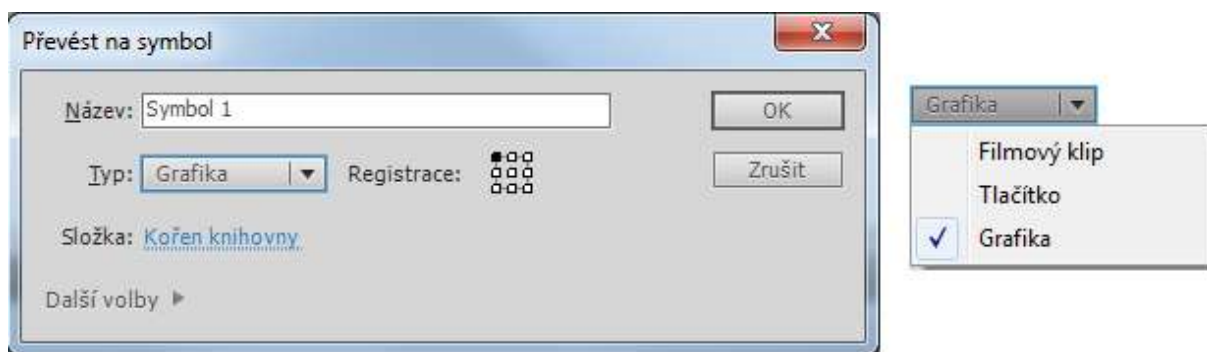
Posledním panelem z popisované skupiny je panel Projekt (obrázek 16.), který umožňuje udržovat informace o jednotlivých souborech vybraného projektu.



Obrázek 16: Panel Projekt

2.5 Základní objekty

Mimo vytvořené základní útvary, existují tři speciální objekty (Symboly), které se používají tak, že umožňují sjednocování základních útvarů a přiřazování jim speciální funkce.



Obrázek 17: Dialogové okno pro převod na Symbol

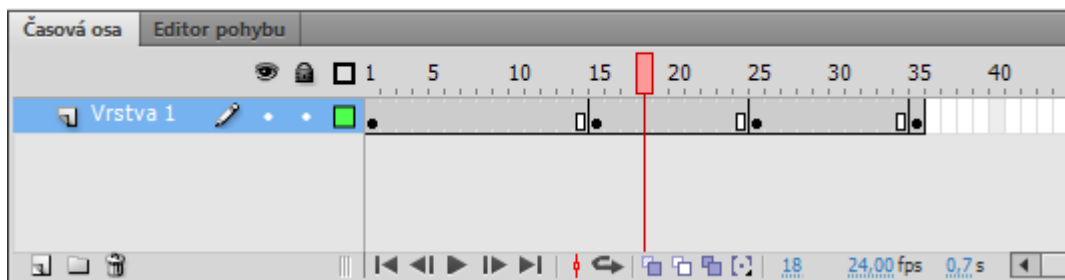
Symboly lze rozdělit na tři typy s různým chováním:

- Filmový klip - symbol obsahující vlastní časovou osu (nezávislost na hlavní časové ose).
- Button - symbol umožňující vytvářet vlastní tlačítka.
- Graphic - symbol umožňující uchovávat obrázky.

Všechny vytvořené symboly lze následně najít v knihovně.

2.6 Časová osa

Jak již bylo zmíněno, je časová osa základní částí celé aplikace. Umožňuje pracovat s jednotlivými snímky a tímto způsobem vytvářet výslednou aplikaci. Umožňuje základní nastavení jednotlivých snímků, které pak ve výsledku vytvářejí výslednou animaci. Vytvářené snímky lze dále třídit podle toho, do které vrstvy jsou vloženy. Výslednou rychlost pak ovlivňuje nastavení frame rate (obvykle 24 snímků/sekundu).



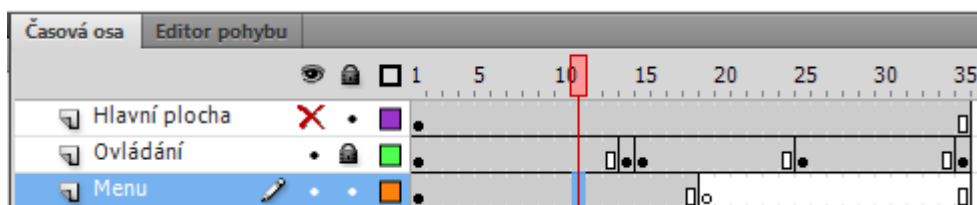
Obrázek 18: Snímky a Klíčové snímky na časové ose

V souvislosti s časovou osou lze použít dva typy snímků:

- **Snímek** - základní součást celé animace podle fps – animace střídá snímky
- **Klíčový snímek** - umožňuje vytvořit výslednou animaci tak, že klíčové snímky určují to, jak se budou počítat jednotlivé Snímky, ležící mezi klíčovými.

Klíčové snímky na časové ose lze odlišit tím, že jsou ve spodní části vidět černé tečky. Na obrázku 18 jsou vidět klíčové snímky na pozicích 1, 15, 25 a 35. Klíčové snímky jsou důležité proto, že určují snímky, na kterých dochází k nějaké klíčové změně. Ostatní snímky ležící mezi těmito klíčovými jsou následně dopočítávané automaticky.

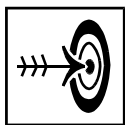
S časovou osou jsou spojeny také vrstvy, které umožňují zpřehlednit výslednou aplikaci, neboť umožňují využít více různých vrstev, pro jednotlivé symboly. Symboly pak mohou reprezentovat jednotlivé části obrazovky. Jednotlivé vrstvy také mohou obsahovat různý počet klíčových snímků a různých pozic (obrázek 19.).



Obrázek 19: Vrstvy na časové ose

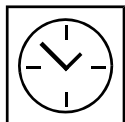
Jednotlivé vrstvy mají mimo jiné vlastní jméno a dále mohou být zobrazeny nebo skryty (můžeme některé vrstvy vypnout, což zpřehledňuje pracovní plochu. Druhou vlastností vrstev je možnost zamčení. Pokud je vrstva zamčená nelze v ní dělat jakékoliv změny.

3 Základní metody doplňování



Cíl kapitoly

Cílem této kapitoly je seznámit zájemce s třemi základními typy doplňování, používaných ve spojení s časovou osou.



Čas na prostudování kapitoly

Čas pro nastudování této kapitoly je asi 120 minut.

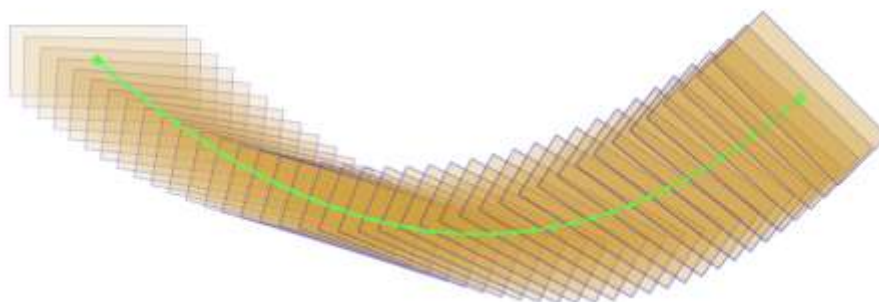
Ručně vytvářet všechny snímky by bylo velmi namáhavé, proto existují ve Flashi tři automatické metody doplňování snímků. Mezi tyto metody patří:

- **Vytvořit doplňování pohybu** - animace pohybu po křivce
- **Vytvořit doplňování tvaru** - animuje tvar symbolů mezi jednotlivými klíčovými snímky
- **Vytvořit klasické doplňování** - animace pohybu pomocí klíčových snímků

3.1 Doplňování pohybu

Toto doplňování umožňuje vytvořit řídící cestu pro symbol. Touto cestou se pak symbol pohybuje tak, aby prošel jednotlivé snímky. Při použití tohoto doplňování vytvoříme klíčový snímek a vybereme požadovaný symbol. Následně vybereme snímek z oblasti, ve které se má provádět animace pohybu a nastavíme pomocí pravého tlačítka a volby doplňování pohybu požadované doplňování.

Následně lze vybraný symbol přesouvat na jednotlivých snímcích a výsledná cesta se bude dopočítávat automaticky. Takto lze nastavovat pohyb, rotaci, velikost a alfa kanál. Ukázka doplňování pohybu je na obrázku 20.



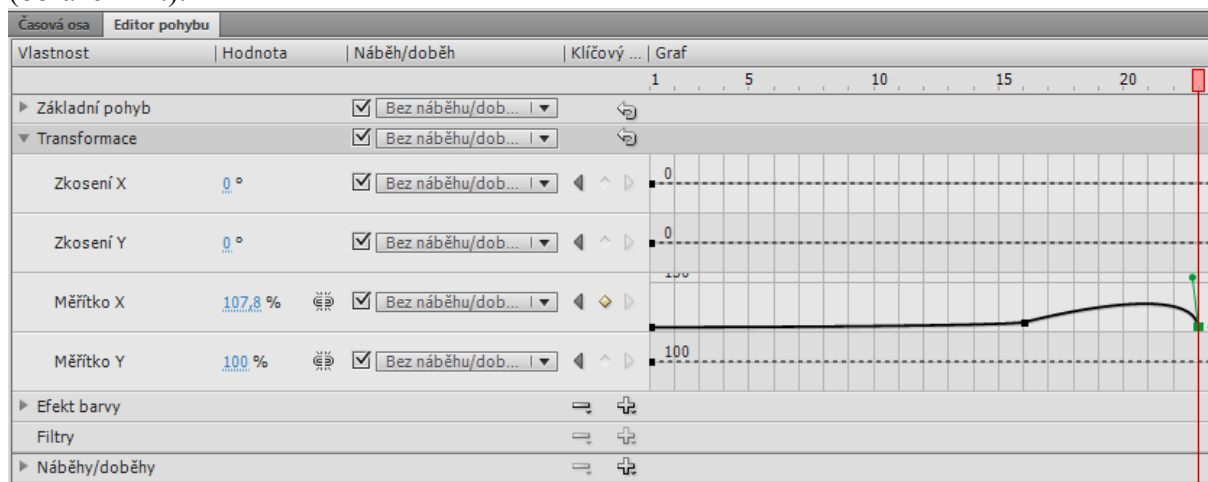
Obrázek 20: Ukázka doplňování pohybu

Celou animaci si lze prohlédnout pomocí Průsvítek, Obrysu průsvítek, úpravy více snímků a nastavení značek. Všechny čtyři ovládací tlačítka jsou na obrázku 21.



Obrázek 21: Nastavení zobrazení více snímků

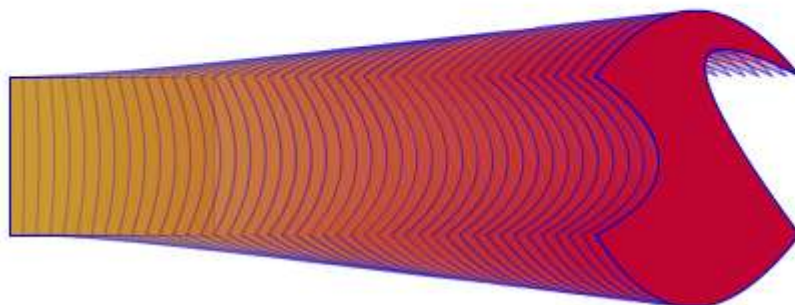
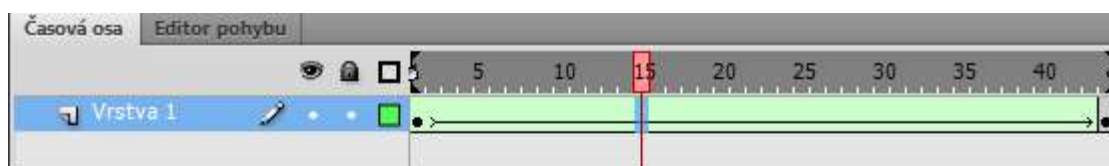
Editor pohybu - tento nástroj umožňuje zobrazit průběh jednotlivých transformací pomocí transformačních křivek, editací jednotlivé křivky u elementárních transformací můžeme opět upravovat jednotlivé průběhy transformace (obrázek 22.).



Obrázek 22: Editor pohybu s editační křivkou pro měřítko na ose X

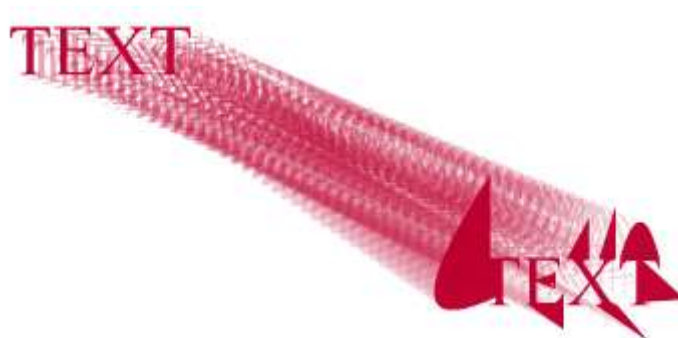
3.2 Doplnění tvaru

Tento typ doplnění umožňuje transformovat tvar objektu. Zdrojem transformace není jako v předešlém případě symbol, ale pouze základní vektorový obrázek. Pokud základní obrázek nekonvertujeme na symbol, nelze již přímo měnit jeho tvar, ale pouze po editaci tohoto symbolu. Pro doplnění tvaru tedy vybereme vektorový obrázek nebo jeho část a vytvoříme doplnění tvaru. Následně na dalším klíčovém snímku můžeme editovat a transformovat tento tvar (obrázek 23.).



Obrázek 23: Ukázka doplnění tvaru

Doplnění tvaru lze použít nejen ke změně tvaru, ale také pro změnu barvy, alfa kanálu apod.



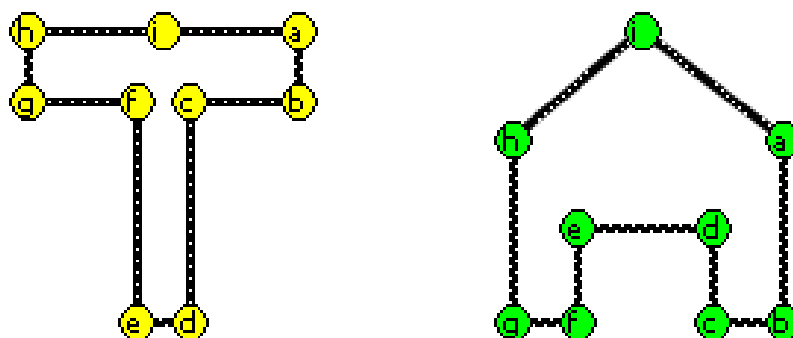
Obrázek 24: Převedení textu na vektory a jejich deformace

Objekty (např. font) je třeba převést na vektorový obrázek pomocí *Rozdělit* (Break Apart, Ctrl+B).

Výslednou transformaci samozřejmě nelze nalézt v Editoru pohybu, neboť se nejedná o transformaci jako v případě Doplnění pohybu. Na druhou stranu umožňuje Doplnění tvaru tzv. řízenou transformaci tvaru pomocí *Pomocných bodů tvarů*.

Pomocné body, které umožňují řídit výslednou animaci

- Přidání pomocného bodů (Ctrl+Shift+H)
- Zobrazení pomocných bodů (Ctrl+Alt+H)



Obrázek 25: Řízení animace s využitím pomocných bodů

Na obrázku 25. je zobrazena animace za pomoci řídicích pomocných bodů a jsme schopni velmi jednoduše řídit výslednou animací. Princip spočívá v tom, že oba klíčové obrazy mají řídicí pomocné body, které definují totožné body výsledné animace.

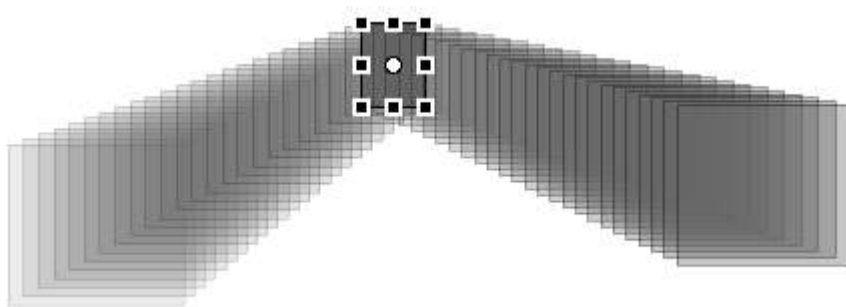
3.3 Klasické doplnění

Tento typ doplnění umožňuje jednoduché transformace (neobsahují řídicí osu pohybu apod.). Doplnění lze provést jak na vektorových obrázcích, tak na symbolech. Je možné jednoduše doplňovat velikost, pozici, rotaci apod.



Obrázek 26: Ukázka časové osy po použití klasického doplnění

Ukázka klasického doplnění je na obrázku 26., kde lze vidět dva typy transformace a to posunutí a změnu velikosti.

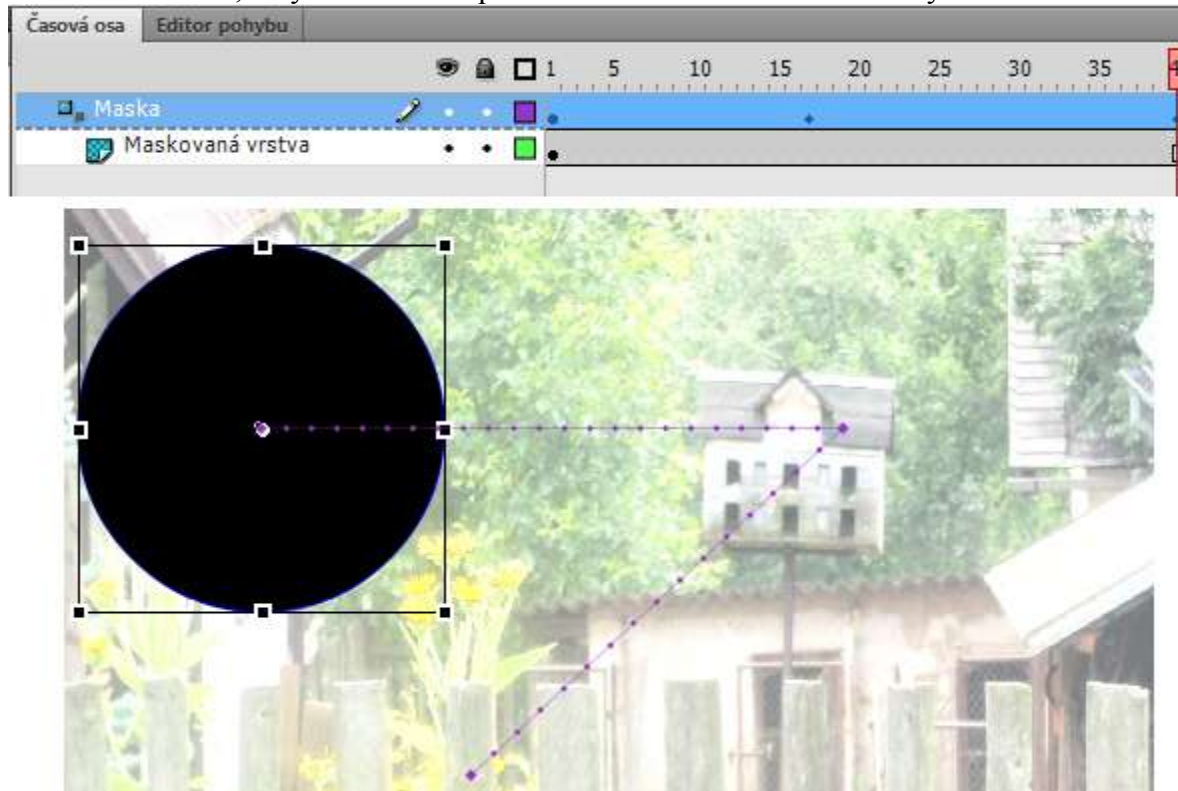


Obrázek 27: Klasické doplnění

3.4 Maska (animace maskou)

Tato animace umožňuje vytvořit základní vrstvu, obsahující například obrázek a dále vrstvu typu maska (obrázek 28.), která obsahuje v tomto případě symbol s doplněním pohybu. Tím dojde ve výsledku k tomu, že po zpuštění a přehrávání bude scéna viditelná pouze v místě masky, neuvidíme tedy celý obrázek ale animovaný pohyb masky a část obrázku, který je v danou chvíli pod zvolenou maskou.

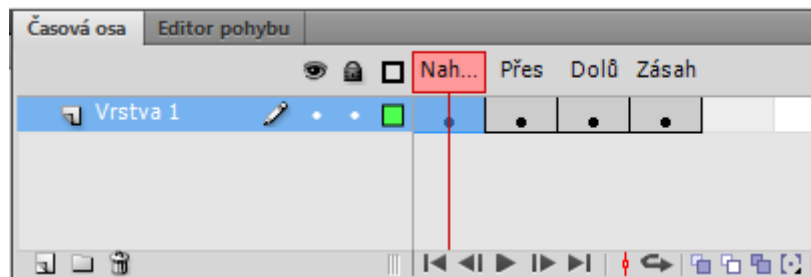
Vytvoření vrstvy s maskou docílíme jednoduše tím, že vytvoříme novou vrstvu ležící nad požadovanou vrstvou, kterou chceme maskovat, a tuto novou vrstvu nastavíme jako *Maska*. Tím docílíme, po vložení nějakého tělesa do této vrstvy, efektu maskování, tedy zobrazování pouze určité části maskované vrstvy.



Obrázek 28: Animace maskou

3.5 Symbol Tlačítko (Button)

Jedním z velmi důležitých symbolů nástroje Flash je symbol *tlačítko*. Tento symbol umožňuje díky svým předdefinovaným vlastnostem vytvářet jednoduše a pohodlně různá tlačítka. Samotný symbol obsahuje čtyři klíčové snímky (obrázek 29.) definující chování objektu.

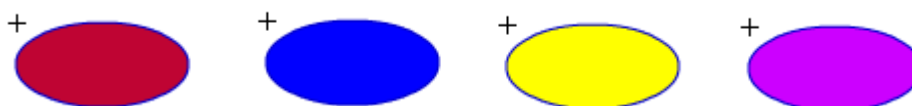


Obrázek 29: Časová osa pro symbol tlačítko

Základní klíčové snímky symbolu tlačítko:

- Nahoře - snímek určuje základní vzhled tlačítka.
- Přes - definuje tvar tlačítka po najetí kurzoru myši.
- Dolů - klíčový snímek definuje tvar tlačítka, pokud uživatel stiskne tlačítko myši.
- Zásah - tento klíčový snímek určuje vlastní aktivní část tlačítka.

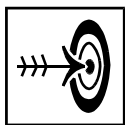
Pomocí tohoto symbolu můžeme lehce vytvářet zajímavé tvary ovládacích tlačítek včetně jejich aktivní části (kruhové tlačítko má jinou aktivní část než oválné nebo obdélníkové apod.) Ukázka všech čtyř snímků je na obrázku 30.



Obrázek 30: Jednotlivé klíčové snímky oválného tlačítka

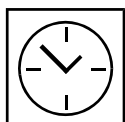
I zde samozřejmě máme v případě složitějšího tlačítka k dispozici jednotlivé vrstvy časové osy, umožňující přehlednit výsledné vytváření symbolu.

4 Vytváření aplikace



Cíl kapitoly

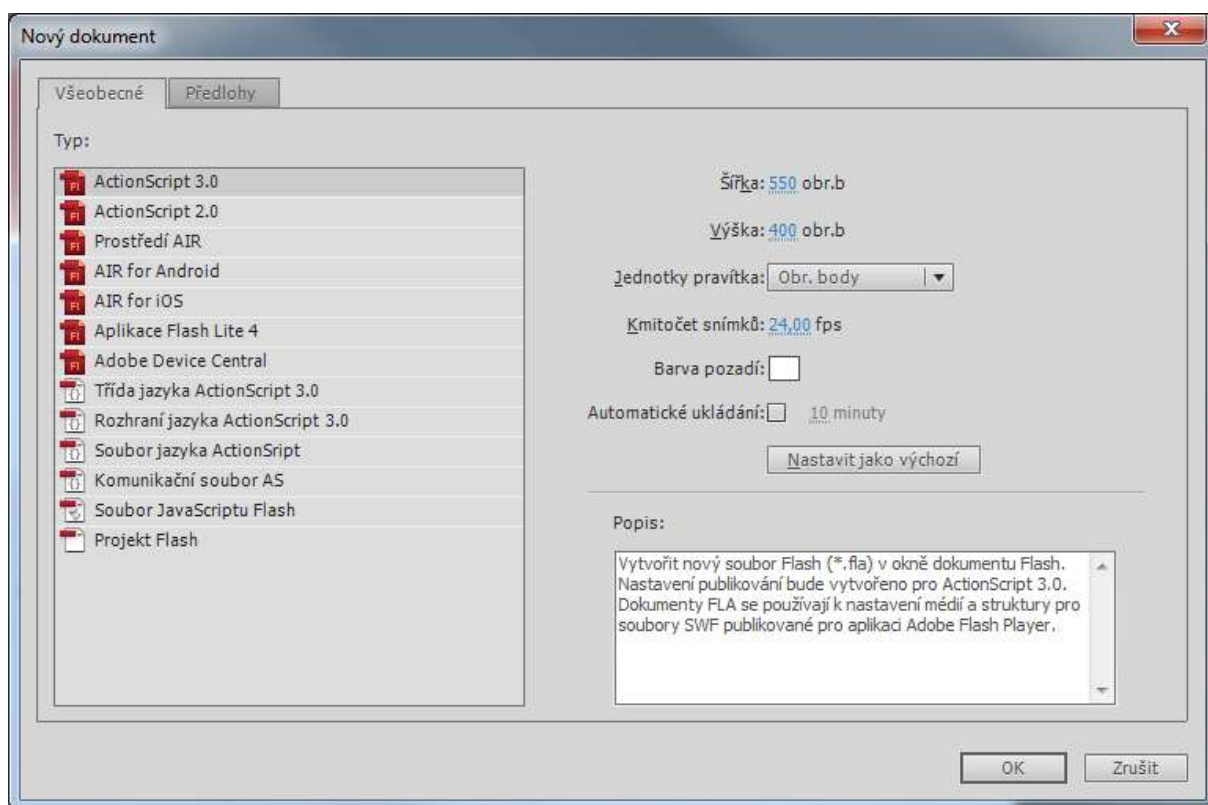
Tato kapitola se zabývá vytvářením aplikace včetně výsledného publikování zdrojového souboru a to jako standalone aplikace nebo jako web aplikaci s příslušnou podporou ActionScriptu dané verze.



Čas na prostudování kapitoly

Čas pro nastudování této kapitoly je asi 120 minut.

Na začátku vytváření aplikace máme možnost zvolit si typ nového dokumentu, zde definujeme, o co se ve výsledku bude jednat, zda standardní nebo web aplikace, popřípadě aplikace pro iOS nebo Android apod. Dále můžeme nastavit základní vlastnosti jako je šířka a výška obrazových bodů, nastavení jednotek, snímku za vteřinu a barvy pozadí (obrázek 31.).



Obrázek 31: Vytváření nového dokumentu

Vybraný typ dokumentu samozřejmě ovlivňuje chování a další možnosti, které bude mít autor při vytváření Flash aplikace.

4.1 Vyšší využití Adobe Flash

S vývojem Adobe Flash přišel posun také s vývojem vnitřního jazyka, využívaného pro vyšší možnosti při vytváření aplikací a jejich složitějších částí. I tento kdysi skriptovací jazyk prošel postupným vývojem a dnešní verze s názvem

ActionScriptu 3 se posunula ve vývoji a možnostech velkým krokem kupředu. Mezi nejzákladnější výhody, které tato verze ActionScriptu přinesla, patří mimo jiné objektově orientované programování (třídy, metody apod.), dále například 3D transformace apod.

Za pomoci ActionScriptu můžeme rozšiřovat stávající animace o další možnosti interaktivity (tlačítka, posuvníky, menu, reakce apod.).

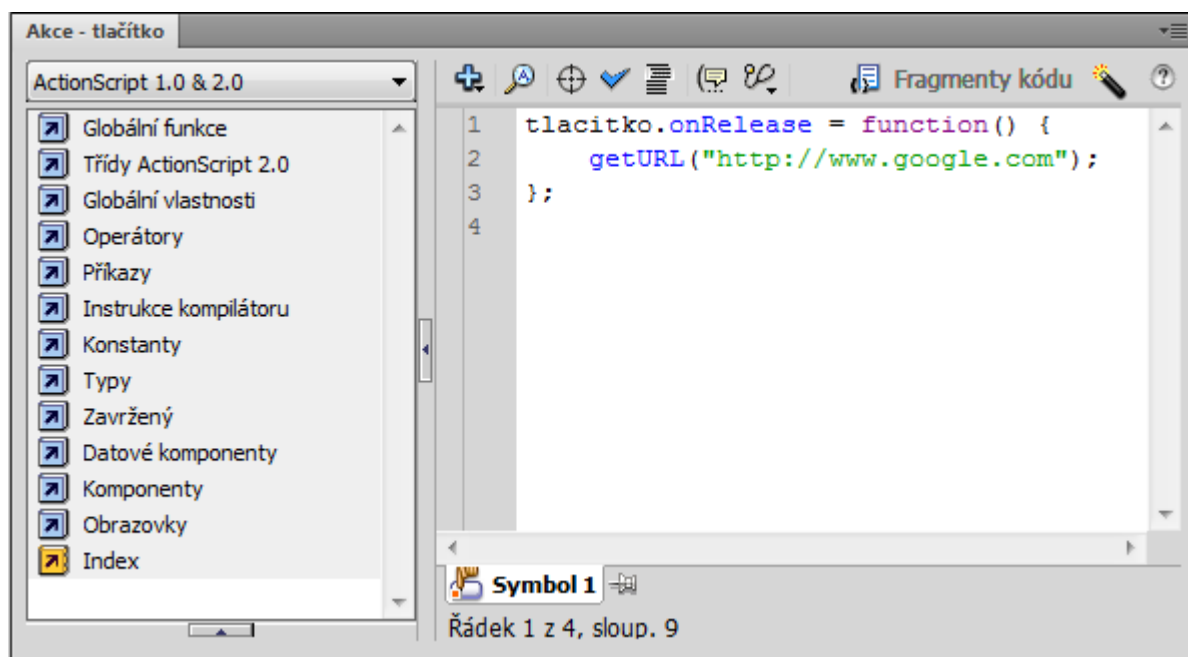


Obrázek 32: Interaktivní prvky Adobe FLashe

4.2 Ukázka vytváření ActionScriptu a porovnání verzí

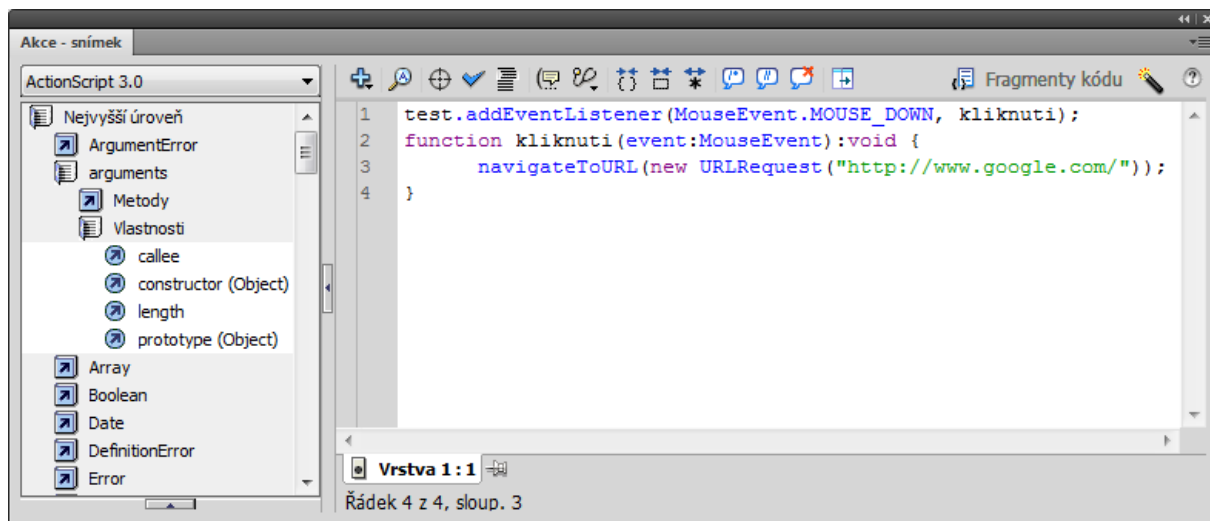
Příklad a srovnání rozdílu obou verzí uvedena na příkladu, kterým bude jednoduché tlačítko mající přiřazenou reakci otevření html stránky www.google.com.

Vytvoření skriptu při použití ActionScript 2.0 vytvoříme symbol tlačítko, kterému následně musíme přiřadit skript pro otevření požadované stránky. Přiřazení skriptu docílíme tím, že klikneme na požadovaný objekt pravým tlačítkem a dáme možnost akce. Následně v nabídce Akce vytvoříme zdrojový kód otevírající danou html stránku po kliknutí.



Obrázek 33: Vytvoření reakce na stisk tlačítka v ActionScript 2.0

V případě použití ActionScriptu 3.0, se již jednotlivé části zdrojového kódu nepřirazují jednotlivým objektům, ale vytváří se jediný zdrojový kód pro celou aplikaci. Ukázka výsledného skriptu je na obrázku 34.

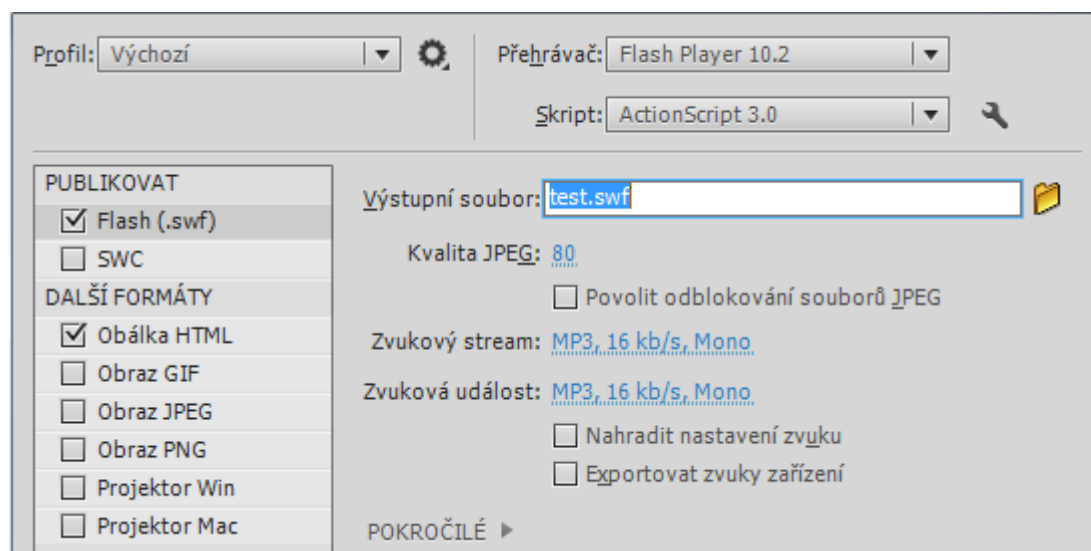


Obrázek 34: Vytvoření reakce na stisk tlačítka v ActionScript 3.0

4.3 Publikování výsledného projektu

Výslednou aplikaci je nutné ze zdrojového souboru publikovat (Alt+Shift+F12). Na výběr je několik formátů (obrázek 35.), včetně jejich konkrétních nastavení (Ctrl+Shift+F12). Výstup nejčastěji bývá do standardní spustitelné aplikace (Windows nebo MAC), příkladem je standalone windows aplikace (soubor s příponou *.exe).

Dalším velmi využívaným výstupem je aplikace pro použití v HTML stránce (jedná se o formát *.swf a *.html). Při vhodném vytvoření aplikace je možné klidně publikovat výslednou aplikaci ve více formátech.



Obrázek 35: Publikování výsledné aplikace

Příklady výsledných formátů:

- Zdrojový soubor (**soubor.fla**) obsahuje zdrojové informace výsledné animace
- Spouštěcí soubor (**soubor.exe**) umožňuje spouštět animaci na platformě windows, nepotřebuje žádný plugin (existuje varianta pro Macintosh)

- Webová aplikace (*soubor.swf*) určený pro prostředí internetu, potřeba Flash Playeru. Soubor se zakomponuje do internetové stránky (nejnovější verze Flash player 10)
- Animovaný GIF (*soubor.gif*) bez možnosti interakce, pouze rotování obrázku

4.4 Online nápověda

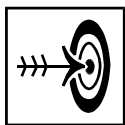
Společnost Adobe má podporu nástroje Adobe Flash velmi silně propracovanou a na stránkách tohoto nástroje [2] lze nalézt velmi rozsáhlý web zpracovávající jednotlivé části včetně online nápovědy.

Součástí nápovědy je samozřejmě i popis jazyka ActionScript a to jak verze 2.0, tak i novější verze 3.0. Mimo jazyk ActionScript lze na stránkách společnosti Adobe nalézt také část věnovanou popisu aplikací pro mobilní zařízení apod.

5 Závěr

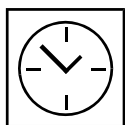
Adobe Flash je dnes velmi silný nástroj pro multimediální prezentace a aplikace jak na webu, tak i jako samostatné aplikace. V poslední době se vývojáři snaží upravit Flash i pro mobilní zařízení. Jeho spojení s novým ActionScriptem 3.0 velmi rozšiřuje také možnosti programování v oblasti programování.

Příloha základních praktických příkladu



Cíl kapitoly

V této kapitole se pomocí několika ukázek seznámíte s praktickými příklady na doplnění pohybu, klasickou animaci apod.



Čas na prostudování kapitoly

Čas pro nastudování této kapitoly je asi 180 minut.

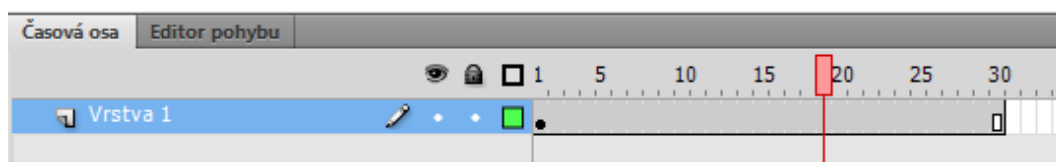
Tato kapitola obsahuje ukázkové příklady, které slouží k ukázce a demonstraci vybraných příkladu.

Příklad na vytvoření doplnění pohybu

Vytvoříme libovolný objekt typu symbol (Tlačítko, Grafika, Filmový klip) na prvním snímku naší časové osy.



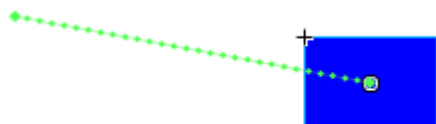
Na časové ose vytvoříme potřebný počet snímků, čímž nastavujeme délku výsledné animace. Délka přehrávání je mimo počtu snímku závislá také na nastaveném počtu snímku za vteřinu.



Kliknutím pravého tlačítka na časové ose a volbou *Vytvořit doplnění pohybu*.

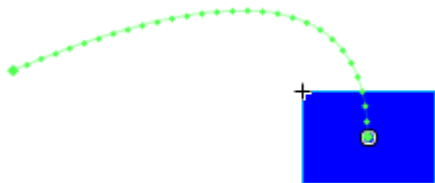


Na posledním snímku (v našem případě snímek 30), který vybereme kliknutím na snímek v časové ose, změníme polohu našeho objektu (symbolu). Symbol se přesune, ale bude již viditelná osa pohybu tělesa.

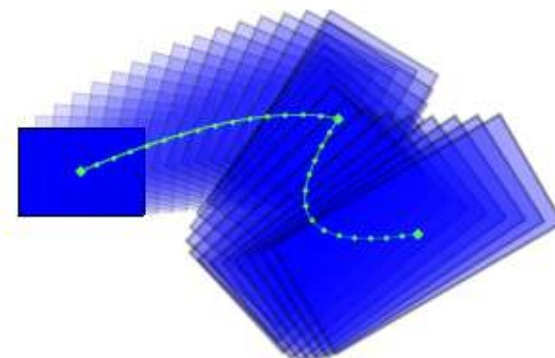


Tvorba animací

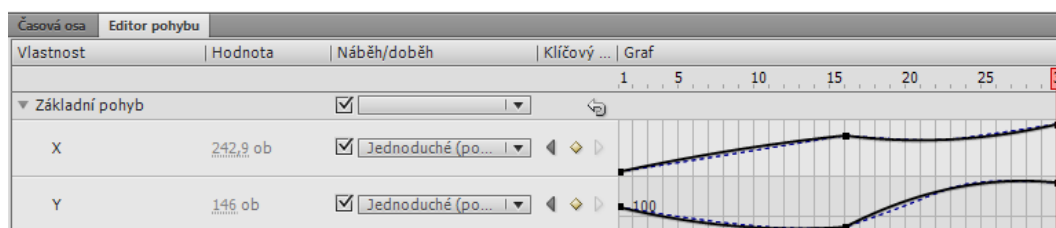
Pozici symbolu můžeme libovolně měnit a ovlivňovat tak výslednou dráhu. Pohyb nemusí být přímočarý, ale lze změnit i tvar dráhy pohybu.



Animace může obsahovat na časové ose i další klíčové snímky. Mimo dráhu pohybu můžeme samozřejmě měnit i další parametry jako je rotace, velikost tělesa apod. Pro lepší přehled výsledné animace můžeme zapnout zobrazování více snímku a vidět tak jednotlivé snímky na celé časové ose.



Jednotlivé transformační křivky lze zobrazit pomocí *Editoru pohybu*, kde je lze dále modifikovat a ovlivňovat tak výslednou animaci.

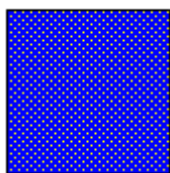


Vytvářenou animaci si lze kdykoli přehrát klávesovou zkratkou Ctrl+Enter. Zdrojová animace je uložena v příloženém souboru *animace_pohybu*.

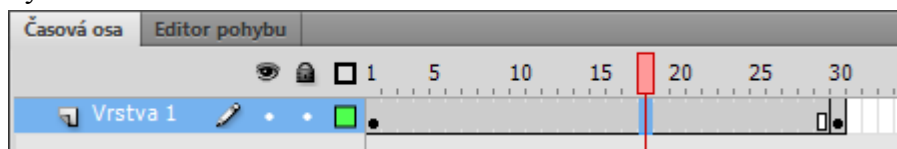
Příklad na vytvoření doplnění tvaru

Pro demonstraci doplnění tvaru budeme opět potřebovat libovolný objekt, aktuálně se nemusí jednat o symbol. Vložíme tedy na prvním snímku obdélník.

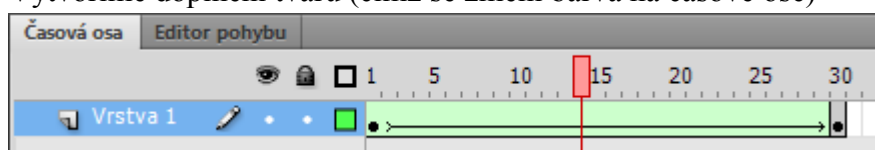
Tvorba animací



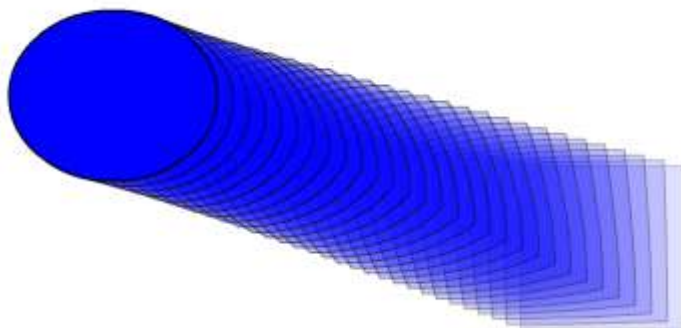
Na časové ose vložíme na libovolný snímek nový *klíčový snímek*. V našem případě byl zvolen snímek číslo 30.



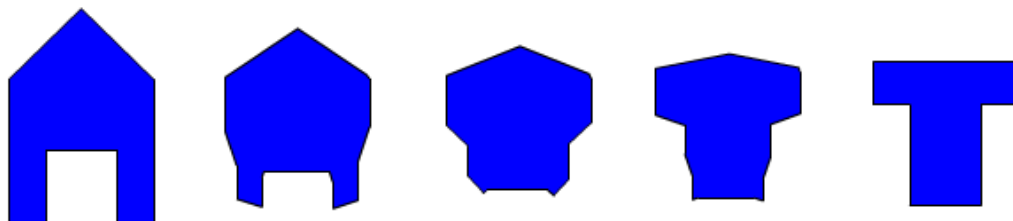
Vytvoříme doplnění tvarů (čímž se změní barva na časové ose)



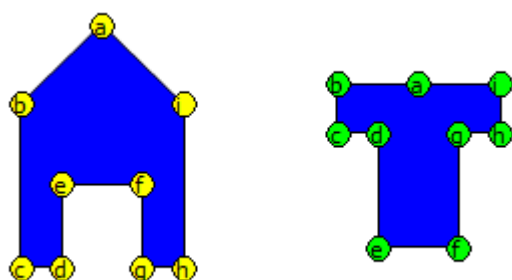
Následně můžeme upravit tvar výsledného tělesa nebo vytvořit jiné těleso. V našem případě smažeme obdélník a vytvoříme elipsu. Po přehrání animace by měl vzniknout efekt přechodu obdélníku na elipsu.



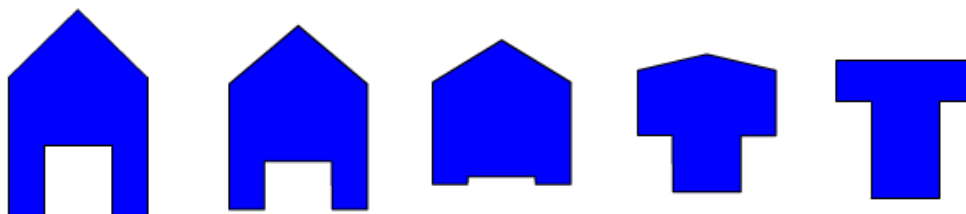
Obdobně lze vytvořit samozřejmě libovolnou animaci, která bude umožňovat změnu tvaru. Při složitějších animacích však nelze nechat animaci probíhat automaticky a je nutné ji řídit. Příkladem takovéto animace může být transformace písmene T na domeček.



Pomocí pomocných bodů, můžeme dosáhnout toho, že nadefinujeme, jednotlivé body mezi vzorem i obrazem (které body se transformují do kterých).



Výsledná řízená animace tak může mít zcela jiný průběh.



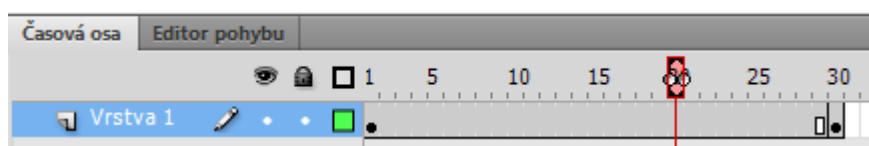
Výsledné ukázky lze nalézt v příložených souborech (doplneni_tvaru)

Příklad na vytvoření klasického doplnění

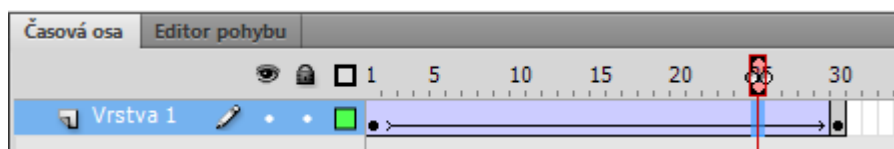
Příkladem klasického doplnění může být změna pozice, velikost rotaci apod. Použitý objekt nemusí být symbol (flash si ho na symbol transformuje). V našem případě se jedná o obdélník.



Vložíme klíčový snímek na časovou osu (v našem případě snímek 30).

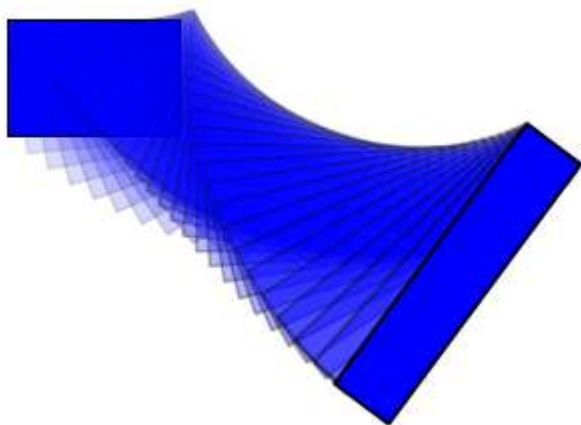


Dále nastavíme klasické doplnění.



Nyní můžeme vytvořit rotaci, změnit velikost a animace bude dopočítána.

Tvorba animací



Transformují se pouze vlastnosti symbolu jako pozice, rotace a velikost. Měnit tvar, barvu apod. nelze.

Použitá literatura

[1] <http://www.adobe.com>

[2] http://help.adobe.com/cs_CZ/flash/cs/using/index.html