

### Vysoká škola báňská – Technická univerzita Ostrava



# Základy Adobe FLASH CS4

# Martin Němec

Ostrava 2011

Název: Základy Adobe FLASH CS4 Autor: Martin Němec Vydání: první, 20101 Počet stran: 19

Studijní materiály pro řešitele projektu Jazyková korektura: nebyla provedena.

#### Určeno pro projekt:

Operační program Vzděláváním pro konkurenceschopnost Název: Personalizace výuky prostřednictvím e-learningu Číslo: CZ.1.07/2.2.00/07.0339 Realizace: VŠB – Technická univerzita Ostrava Projekt je spolufinancován z prostředků ESF a státního rozpočtu ČR

© Martin Němec © VŠB – Technická univerzita Ostrava

# Obsah

Úvod 4
Postup při tvorbě flash aplikace
3arvy a jejich použití
Časová osa9
Notion Tween (Doplnění pohybu)9
Shape Tween(dopInění tvaru)
Classic Tween (klasické doplnění)11
Vaska (animace masek)
Γlačítko (Button)
√ýstup programu Flash13
Skriptování v ActionScriptu
Online nápověda14
Praktické ukázky 14
Závěr19
Použitá literatura

## Úvod

Adobe Flash je velmi silný nástroj pro tvorbu interaktivních multimediálních animací, který lze použít jak pro samotné aplikace (dodané např. na DVD) tak zejména pro webovské prezentace. Mezi základní výhody patří možnost vektorového formátu, což umožňuje minimalizovat výslednou velikost aplikace, což je jedním z velkých požadavků při tvorbě www stránek. Flash je vhodný zejména pro tvorbu aplikací s různými zvukovými a video efekty.

Srovnávání technologií

- Animovaný GIF (lze označit za předchůdce), umožňuje pouze statické přehazování obrázků
- Adobe Flash jednoduché vytváření interaktivních prezentací a animací. Mezi nevýhody patří potřeba instalace Flash playeru pro přehrávání v prohlížečích (vše volně dostupné na webu)
- JavaApplet potřeba znalosti jazyku Java (náročnější), potřeba instalace Javy
- Další možnosti (JavaSript, Silverlight apod.)

Co vše Flash umožňuje?

- Jednoduše vytvářet rozsáhlé interaktivní animace.
- Využití vektorového formátu, což umožňuje velmi omezit v některých případech výslednou velikost animace. To je velmi důležité zejména při použití ve web prohlížečích.
- Základem časová osa, umožňující sestavovat animace pomocí různých efektů (posun, rotace, změna velikosti, tvarů, barvy, textu apod.).



• Za pomocí ActionScriptu jednoduše rozšířit tyto animace o interaktivitu (tlačítka, posuvníky, menu, reakce apod.).



Mezi základní výhody patří:

dostupnost bezpečnost jednoduchost vektorové animace velikost práce s videem, zvukem interaktivita použití lokálně i na internetu

### Postup při tvorbě flash aplikace

Základní postup při tvorbě flash aplikace lze rozdělit do části kde se kreslí objekty, dále se tvoří jejich animace a přidává interaktivita (doplnění o skripty).

Výslednou aplikaci lze uložit v následujících základních formátech:

- Zdrojový soubor (soubor.fla) obsahuje zdrojové informace výsledné animace
- Spouštěcí soubor (*soubor.exe*) umožňuje spouštět animaci na platformě windows, nepotřebuje žádny 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

Pro návrh výsledné aplikace

Fl	File Edit View Insert Modify Text Commands Control Debug	Wir	idow Help		CLASSIC 🔻		
44	TIMELINE MOTION EDITOR		Duplicate Window 0	Ctrl+Alt+K			**
<b>R</b> <b>R</b> <b>R</b>	(1) (2) (	✓	Toolbars Timeline ( Motion Editor	► Ctrl+Alt+T	70 75 80 85 90	FL Document Untitled-1	*=
0		$\checkmark$	Tools	Ctrl+F2		PUBLISH	
9.		-	Properties	Ctrl+F3		Player: Flash Player 10	
9 A			Library	Ctrl+L	0	Script: ActionScript 3.0	
8,			Common Libraries	+		Classes .	
1	□ □ 🛱 🕴 🛉 🖿 🖓 🔂 1. 24.0 fps 0.0 s 🔍		Motion Presets			class.	
	Untitled-1* ×		Actions	F9	Ait .	Profile: Default Edit.	
	🔶 🚝 Scene 1		Behaviors	Shift+F3	🖆 \land 100% 🔹 📷	AIR Settings Edit.	
1			Compiler Errors	Alt+F2	▲ マ	PROPERTIES	
119			Debug Panels	+		FPS: <u>24.00</u>	
100			Movie Explorer	Alt+F3		Size: 550 x 400 px Edit.	
5.			Output	F2			
S.			Align	Ctrl+K		Stage:	
0			Color	Shift+F9			
<u>2</u>			Info	Ctrl+I	Animator		
107			Swatches	Ctrl+F9	Classic		
4			Transform	Ctrl+T	Designer		
2			Components	Ctrl+F7	Developer		
			Component Inspector	Shift+F7	Essentials		
<b>1</b>			Other Panels	•	Prost (Classic)		
5			Extensions	•	New Workspace		
<u>.</u>			Workspace	•	Manage Workspaces		
			Hide Panels	F4			
*5		-	1 Untitled-1				
*		_			· · · · · · · · · · · · · · · · · · ·		
					-		
	۲ III				•		

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

Panel nástrojů umožňuje následující možnosti:

Selection Tool (V) - posunování s objekty, editace již vytvořených objektů, přesouvání 8 8 Subselect Tool (A) - editace již vytvořených čar, objektů a křivek Free Transform Tool (Q) - změna polohy a vzhledu objektů Rotare and Skew - změnu rotace, ale také můžeme změnit zkosení objektu  $^{\circ}$ Scale - změna velikosti zvoleného objektu \$. Т Distort - změnit tvar rohů u zvoleného objektu -Envelope - pomocné body umožňující dokonalejší editaci tvaru 1 3D Rotation Tool (W) - umožňuje 3D rotace s objekty Ì Lasso Tool (L) - slouží k přesnějšímu výběru objektů i jejich částí Pen Tool (P) - slouží k vytváření objektů pomocí klíčových bodů S. O. Text Tool (T) - vkládání textů a jejich editace ¥ 🛛 Line Tool (N) – vytváření úseček (SHIFT - rovné úsečky) Rentangle Tool (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 Oval Tool (O) - vytváření kruhů a oválů - - - > Pencil Tool (Y) - kreslení úseček od ruky. Straighten (ostré zarovnání), Smooth A \*5 (oblé zarovnání), Ink (neupravuje nic) Brush Tool (B) - kreslení křivek, které připomínají tah štětce. Deco Tool (U) - dekorativní motivy Bone Tool (X) - výběr objektů Paint Bucket Tool (K) - vyplnění objektu danou barvou či gradientem Ink Bottle Tool (S) - barva ohraničení Eyedropper Tool (I) - zjištění barvy daného objektu. Následně po vybrání se aktivuje Paint **Bucket Tool** Eraser Tool (E) - mazání jednotlivých objektů či částí, které jsou umístěny ve scéně klipu Hand Tool (Q) - přesunování celého dokumentu Zoom Tool (M) - zvětšování a zmenšování dokumentu v procentech Mezi základní objekty patří následující Line Tool(N) Oval Tool (O) Rectangle Tool (R) Pencil Tool (Y)

- Brush Tool (B)
- Pen Tool (P)



Kombinace základních objektů

Jednotlivé objekty a jejich částí lze slučovat do skupin nebo zpětně rozkládat. Pro slučování vybraných částí

**Group** (Ctrl+G) - vytvoření skupiny z vybraných objektů **Ungroup** (Ctrl+Shift+G) – zrušení skupiny



Vytvářené objekty je samozřejmě potřeba editovat. Pro editaci hranice jednotlivých objektů máme k dispozici několik možností editace

Možnost vytváření křivky z hrany, s přidrženým ALT vytvářet další vrcholy na místě kliknutí Možnost editovat hrany a křivky

Možnost nastavovat velikost, zkosení, rotaci (kolečko střed)



#### Barvy a jejich použití

Míchání barev (COLOR)

None - žádná výplň. Solid - základní výplň celého objektu Linear Gradient - vyplní objekt lineárně dvěma (nebo i více) barvami. Radial Gradient - výplň odpovídá tvaru kruhu.

#### Bitmap - použití rastrového obrázku (textura)



Panel pro okamžitou změnu barvy (SWATCHES) předdefinované barvy (i Lineár a Radial Gradient)

Jednou z dalších možností je zarovnávání několika vybraných objektů a nastavování jejich vlastností.

a

A

	44	
		ALIGN INFO TRANSFORM
	8	↔ 100.0 % \$ 100.0 % © أ
		⊙ Rotate
		△ 0.0 °
ALIGN INFO TRANSFORM		O Skew 27 0.0 ° ♀ 0.0 °
·····································	0	3D Rotation
Distribute: stage:	111	X: <u>-360.0</u> ° Y: <u>258.0</u> ° Z: <u>0.0</u> °
		3D Center point
Match size: Space:		X: <u>386.2</u> Y: <u>209.6</u> Z: <u>0.0</u>
		Ge 12
_+ W: 20.0 + + X: 40.0		
H: 20.0 Y: 40.0		
R: -		
B:- Y: 459.0		
A: -		

Mimo vytvořené základní útvary, existují tři speciální objekty (Symboly, F8)

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

### Časová osa

Umožňuje základní nastavení jednotlivých snímků, které pak ve výsledku vytvářejí výslednou animací. Vytvářené snímky lze dále třídit podle toho, do které vrstvy jsou vložené. Výslednou rychlost pak ovlivňuje nastavení frame rate (obvykle 24 snímků/ sekundu).



V souvislosti s časovou osou lze použít dva typy snímků:

Frame (snímek) – základní součást celé animace podle fps – animace střídá snímky Keyframe (klíčový snímek) umožňuje vytvořit pouze klíčové snímky zbytek se dopočítává

Ručně vytvářet všechny snímky je velmi namáhavé, proto existují ve Flashi tři automatické animace:

Motion Tween - animace pohybu po řídící křivce Shape Tween - animace tvaru Classic Tween - animace pohybu pomocí keyframe

#### Motion Tween (Doplnění pohybu)

Program Adobe Flash umožňuje uživateli nastavit pouze některé řídící snímky na časové ose a zbývající snímky dopočítává Flash samostatně. Mezi první doplnění patří doplnění pohybu. K těmto účelům musí být základní objekt, na který chceme doplnění pohybu použít konvertován na symbol, kterým pak můžeme pomocí řídící křivky pohybovat.

Útvar musí být konvertován na objekt např. na Graphic (F8)

TIMELINE	OUTPUT	CON	MPIL	ER E	RRO	RS	мот		TOR	ACTION	١S					
		9			10	5		10	15	20	25	30	35	40	45	5
📕 🖉 Ani	mace 🍃	۶.			•											Ţ
🕤 ovi	adani				a ••											d

Lehké nastavení animace (pohyb, rotace, velikost, alfa kanál)



Motion editor – klíčové snímky nemusí být pro složitější animaci dostačující, proto je součástí programu Flash i Motion editor (editor pohybu), který umožňuje pomocí křivek zobrazovat a měnit průběhy jednotlivých transformací.

TIMELINE OUTPUT	COMPILER ERRORS	MOTION EDITOR	ACTIONS			
Property	Value	Ease	Keyframe	Graph		
				5 10 15 20	25	30 35 40
Basic motion		No Ease	(p			
x	<u>112 px</u>	No Ease	۱ 🔶 ک		•	
Y	145.5 px	No Ease	۱ 🗘 🖓 🕨			Remove Keyframe
Rotation Z	<u>0</u> °	No Ease	▼	- 0	-	
Transformation		No Ease	<u>و</u> ک 💌		•	Roving
▼ Color Effect			그 산			Corner point
▼ Alpha		No Ease	। 🔹 🤄			Smooth point
20	50 🛄 49			4		Linear left

#### Shape Tween(doplnění tvaru)

Druhým použitelným doplněním v prostředí programu Flash je doplnění tvaru. Toto doplnění umožňuje na jednotlivých klíčových snímcích mít různé tvary a program Flash se automaticky pokusí přechod mezi jednotlivými tvary dopočítat.

Pro animaci může být zvolen libovolný vektorový útvar.

TIMELINE	OUTPU	r	CON	IPIL	ER E	RROF	۲S	MOTI	ON ED	ITOR	ACTIO	NS - FRA	ME					
			9			1	5	1	10	15	20	25	30	35	4	40	45	50
🕤 Ani	mace		+	•		0.												→.
J Ov	adani	Ĩ	•	•		a.												

Animace mezi prvním a koncovým klíčovým snímkem, nastavení animace (změna tvaru, barva apod.)



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

Protože automatické vytvoření animace nemusí být ve výsledku správné, existují ve Flashi pomocné body (Shape hints), které umožňují řídit výslednou animací tím, že na jednotlové částí vzoru i obrazu umísťujeme jednotlivé pomocné body. Tím definujeme konkrétně, který bod vzoru odpovídá vzoru obrazu.



#### Classic Tween (klasické doplnění)

Posledním doplněním je klasické doplnění, které je zaměřeno zejména na změnu základních vlastnosti jednotlivých objektů, jako je rotace, změna měřítka, pozice apod.). Pomoci tohoto doplnění máme k dispozici v podstatě všechny druhy transformací od změny pozice po křivce, změny barvy, změny tvaru, velikosti, rotace a měřítka.

Pro animaci může být zvolen objekt (konverze F8).

TIMELINE	OUTPU	Г	CON	IPIL	ER E	RRO	RS	MOT	ION E	DITOR	ACTION	NS - FRA	AME				
			9			1	5		10	15	20	25	30	35	40	45	5
n Ani	mace	Ż				•• >						→ <b>•</b> ≻					<b>→ •</b>
J Ovl	adani			+		a.											П



nastavení animace (velikost, pozice, rotace)

Složitější animace se samozřejmě ve výsledku mohou skládat ze všech třech animací. Výsledná složitější animace tak vzniká kombinací všech třech doplnění.

#### Maska (animace masek)

Další velmi používanou technikou v programu Flash je využití animace masky. Ta umožňuje různě kombinovat dynamicky jednotlivé vrstvy nebo jejich řástí.

Vytvoříme vrstvu s maskou nad maskovanou vrstvou a nastavíme ji vlastnost Mask





Výsledná animace pak bude obsahovat pouze tu část obrazovky, kterou bude překrývat černý ovál, na který je navíc nastaveno doplnění pohybu, což znamená, že se elipsa bude pohybovat, přitom bude odkrývat tu část obrázku, nad kterou se zrovna elipsa nalézá.

#### Tlačítko (Button)

Jedním ze základních operací ve flushi bude vytváření tlačítek, ať již ve smyslu opravdových ovládacích tlačítek nebo částí obrazovky reagující na nějakou akci myši. Pro vytvoření vlastního tlačítka využijeme objektu typu symbol a konkrétně symbolu tlačítko.

	op Over	Down  Hit	
🛛 🕼 Layer 1 🖉 🛀		• •	

Po vytvoření tlačítka bude symbol obsahovat čtyři základní snímky. Tyto čtyři snímky v sobě skrývají následující reakce

- jak vypadá tlačítko normálně,
- jak vypadá při najeti myší,
- při stisku tlačítka na myši,
- oblast reakce myši.

#### Výstup programu Flash

Výstup nejčastěji \*.exe nebo \*.swf a \*.html , výsledný soubor nelze rozebrat tak lehce jako HTML kód.

Publish Settings		X
Current profile: Default	▼ ± + ÷	0
Formats Flash HTML		
Type:	File:	
✓ Flash (.swf)	Untitled-1.swf	
✓ HTML (.html)	Untitled-1.html	
GIF Image (.gif)	Untitled-1.gif	
JPEG Image (.jpg)	Untitled-1.jpg	
PNG Image (.png)	Untitled-1.png	10
Windows Projector (.exe)	Untitled-1.exe	
Macintosh Projector	Untitled-1.app	
	Use Default Names	
Publish	ок с	ancel

#### Skriptování v ActionScriptu

S vývojem Adobe Flash přišel posun také s vývojem vnitřního jazyka potřebného pro skriptování a vývoj složitějších částí. S příchodem nového ActionScriptu 3 se posunul vývoj velkým krokem kupředu. Mezi nejzákladnější výhody patří objektově orientované programování (třídy, metody apod.), dále například 3D transformace apod.

Příkladem je tvorba tlačítka a jeho reakce

```
ActionScript 2.0
```

Vytvoříme objekt a tomuto objektu přiřadíme skript

```
test.onRelease = function() {
```

getURL("http://www.google.com");

```
};
```

In ActionScript 3.0

Vytvoříme objekt a na snímku už je jediný skript

test.addEventListener(MouseEvent.MOUSE\_DOWN, kliknuti);

function kliknuti(event:MouseEvent):void {

navigateToURL(new URLRequest("http://www.google.com/"));

}

#### Online nápověda

Společnost Abode má podporu nástroje Adobe Flash velmi silně propracovanou a na stránkách tohoto nástroje [2] lze nalézt velmi rozsáhlý web zpracovávající jednotlivé částí 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.

#### Praktické ukázky

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. Dálka přehrávání je mimo počtu snímku závislá také na nastaveném počtu snímku za vteřinu.

Časová osa	Editor pohybu									
			9	۵	5	10	15	20	25	30
🕤 Vrstv	/a 1	1	• •					T		

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

Časová osa	Editor pohybu							
		9 🔒	<b>1</b> 5	10	15	20	25	30
🔎 Vrstv	va 1	2 • •	<b>-</b>					+
								T

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.



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.

Časová osa	Editor pohybu									
Vlastnost		Hodnota	Náběh/doběh	Klíčový	Graf					
					1	5	10	15	20	25 3
▼ Základní	pohyb		▼	ŝ						
x		<u>242,9</u> ob	☑ Jednoduché (po… । ▼	4 ♦ ▷						
Y		<u>146</u> ob	☑ Jednoduché (po । ▼	4 ♦ ▷	<b>100</b>					

Vytvářenou animaci si lze kdykoli přehrát klávesovou zkratkou Ctrl+Enter. Zdrojová animace je uložena v přiloženém 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.



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

Časová osa	Edito	r poh	ybu									
			9		1	5	 10	15	20	25	30	
n Vrstv	/a 1	2	•	•	•						•	

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 přiložených 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.



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

#### Závěr

Adobe Flash je dnes velmi silný nástroj pro multimediální prezentace a aplikace na webu i jako samostatné aplikace. Jeho spojení s novým ActionScriptem 3.0 velmi rozšiřuje také možnosti programování v oblasti programování.

### Použitá literatura

- [1] http://www.adobe.com
- [2] http://help.adobe.com/cs\_CZ/flash/cs/using/index.html