

Programování grafických aplikací (BI-PGA), Přednáška č. 1

Základy programování grafických aplikací, historie, Metodika programování pluginů I

Jiří Chludil

Fakulta informačních technologií
České vysoké učení technické v Praze
<https://courses.fit.cvut.cz/BI-PGA/>



ZS 2022/2023

Obsah přednášek

- 1 Jak rozšířit funkcionalitu 2D a 3D grafických aplikací pomocí
 - ▶ Skriptů
 - ▶ Zásuvných modulů
- 2 Jak fungují vybrané grafické postupy
 - ▶ Transformace
 - ▶ Konvoluce
 - ▶ Osvětlovací modely
 - ▶ Textury
 - ▶ a mnoho dalšího

Obsah cvičení

Prakticky se naučit

- jak naprogramovat zásuvný modul do GIMPu (Photoshopu)
- jak naprogramovat zásuvný modul do Blenderu
- jak naprogramovat aplikaci do prostředí SAGE 2

Sylabus

- 1 Základy programování grafických aplikací, historie, metodika programování pluginů I.
- 2 2D grafika - GIMP: Metodika programování pluginů II, pokročilé funkce.
- 3 2D grafika - GIMP: Výpočetní metody, konvoluce, filtry.
- 4 2D grafika - GIMP: Metody zpracování obrazu.
- 5 3D grafika - Blender: Úvod, metodika, využití jazyka Python, programování pluginů v jazyce Python.
- 6 3D grafika - Blender: Pokročilé transformace modelů, formáty, animace a komunikace s externími aplikacemi.
- 7 3D grafika - Blender: Pokročilé osvětlovací modely.
- 8 3D grafika - Blender: Texturovací moduly.
- 9 3D grafika - Blender: Realistické metody texturování.
- 10 3D grafika - Blender: Datové struktury a algoritmy pro grafiku.
- 11 Zvaná přednáška

Doporučená literatura

- Jiří Žára, Bedřich Beneš, Jiří Sochor, Petr Felkel: *Moderní počítačová grafika (2. vydání)*. Computer Press, 2005.
- GIMP API <https://developer.gimp.org/api/2.0/>.
- Blender API <https://docs.blender.org/api/current>.

Hodnocení – Zápočet

- Semestrální úlohy
 - ▶ 1-N Rozšíření 2D Editoru (Gimp)
 - ▶ 1-N Rozšíření 3D Editoru (Blender)
 - ▶ úlohy mají různá bodové hodnocení dle obtížnosti
 - ▶ maximálně 70 bodů
- Prezentace úloh v SAGELab
 - ▶ použité algoritmy, způsob řešení
 - ▶ na 12. cvičení
 - ▶ maximálně 10 bodů
- Udělení zápočtu je podmíněno splněním 2 podmínek:
 - ▶ získáním minimálně 35 bodů ze semestrální úlohy (minimálně po 1 úloze z kategorie 2D a 3D editoru)
 - ▶ získáním minimálně 5 bodů z prezentace
 - ▶ získáním minimálně 40 bodů celkově.

Hodnocení – Zkouška

Ústní zkouška s písemnou přípravou

- Základní principy rozšiřování grafických aplikací
- Přehled grafických algoritmů.
- Maximálně 20 bodů, minimálně 10 bodů.

Jak lze rozšířit funkcionalitu aplikace

- 1 Makrem
- 2 Scripty
- 3 Přidáním knihovny
- 4 Zásuvným modulem

Rozšíření pomocí maker

Definice

Makro (Makroinstrukce) - sekvence příkazů reprezentované jediným příkazem

- + Jednoduché použití.
- + Urychlení práce.
- + Nevyžaduje schopnost programování.
- + Nevyžaduje kompilaci rozšiřované aplikace.
- Obtížně parametrizovatelné.
- Nerozšiřuje funkcionalitu.

Rozšíření pomocí skriptování

Definice

Skript - interpretovaný programovací jazyk

- + Jednoduché použití.
- + Vyžaduje jen základy programování.
- + Nevyžaduje kompilaci rozšiřované aplikace.
- + Optimalizovaná funkcionlita skriptu (pro rozšiřovanou aplikaci)
- Vyžaduje znalost skriptovacího jazyka a rozhraní rozšiřované aplikace.
- Obtížně parametrizovatelné.
- Omezeně rozšiřuje funkcionlitu.
- Omezené možnosti debugování.

Rozšíření pomocí knihovných funkcí

Definice

Knihovna - software rozšiřující funkcionalitu aplikace s nutností její kompilace

- + Velmi dobrá rozšiřitelnost funkcionality.
- + Dobře debugovatelné.
- + Lze precizně definovat uživatelské rozhraní modulu.
- Obvykle vyžaduje použití stejného jazyka resp. použití API
- Vyžaduje přístup ke zdrojovým kódům
- Vyžaduje kompilaci rozšiřované aplikace.
- Vyžaduje vyšší znalost programování.
- Potřeba vývojového prostředí.
- Vyžaduje dobrou znalost použitého programovacího jazyka a rozhraní rozšiřované aplikace.

Rozšíření pomocí zásuvných modulů

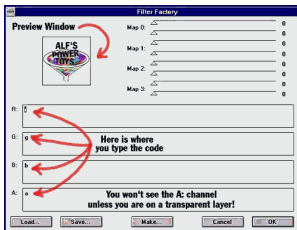
Definice

Zásuvný modul - software rozšiřující funkcionality aplikace bez nutnosti její kompilace

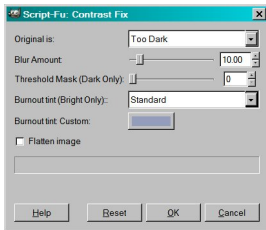
- + Nevyžaduje kompilaci rozšiřované aplikace.
- + Lze integrovat do uživatelského rozhraní rozšiřované aplikace.
- + Obvykle je možné využít více programovacích jazyků.
- + Dobře debugovatelné.
- + Lze precizně definovat uživatelské rozhraní modulu.
- + Velmi dobrá rozšiřitelnost funkcionality (zavisí na API).
- Vyžaduje vyšší znalost programování.
- Potřeba vývojového prostředí.
- Vyžaduje dobrou znalost použitého programovacího jazyka a rozhraní rozšiřované aplikace.

Rozšíření pomocí kombinací základních přístupů

- zásuvný modul umožňující skriptování.

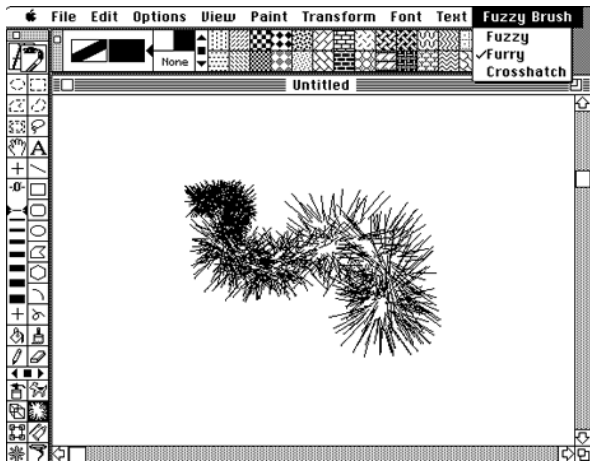


- implementace zásuvného modulu pomocí skriptovacího jazyka



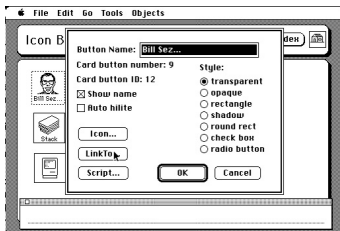
Historie rozšiřování aplikací pomocí pluginů

- okolo 1970 - EDT Textový editor - OS Unisys (UNIVAC)
- 1986 - SuperPaint - OS Mac

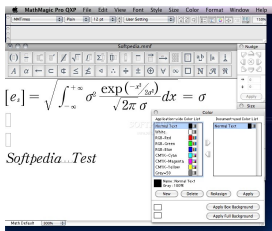


Historie rozšiřování aplikací pomocí pluginů (pokr.)

- 1987 - HyperCard - OS Mac



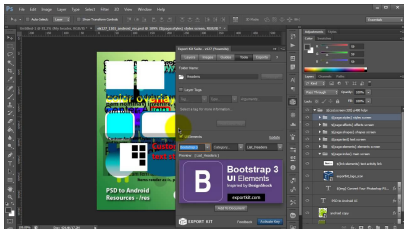
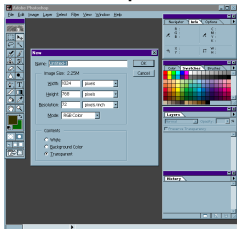
- 1987 - QuarkXPress - OS Mac



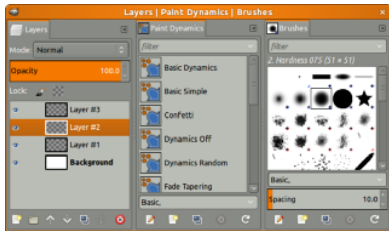
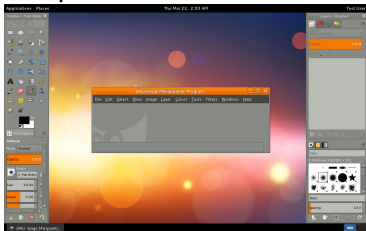
Historie rozšiřování aplikací pomocí pluginů (pokr.)

A co bitmapové editory?

- Photoshop



- Gimp



Historie rozšiřování aplikací pomocí pluginů (pokr.)

A co bitmapové editory? Adobe Photoshop

- 1990 - Verze 1.0 - OS Mac (1996 - OS Windows)
- 1991 - Součástí verze 2.5 je rozhraní pro zásuvné moduly typu filter, import, export a format
- 1994 - Verze 4.0 přidány moduly Výběr barvy a selekce
- 1996 - Verze 5.0 přidán modul automatizace (makro)
- 2000 - Verze 7.0 vylepšený náhled
- 2002 - Podpora velkoformátů

Historie rozšiřování aplikací pomocí pluginů (pokr.)

Gimp

- 1996 - Verze 0.54 - OS Mac (1996 - OS Windows)
- 1996 +6m - API pro zásuvné moduly (formát souborů, filtry)
- 1997 - Verze 0.99 - nové API, podpora skriptů a maker, odděleno GTK+
- 2004 - Verze 2.0 - v principu přepsáno do C++

Co je třeba řešit u zásuvných modulů

- Typ modulu
- Nastavení prostředků (konfigurace)
- API rozšiřované aplikace
- Životní cyklus modulu
- Uživatelské rozhraní modulu
- Management paměti
- Použité matematické a grafické funkce

Typy modulů u bitmapových editorů

- Filtry
- Import
- Export
- Formát
- Výběr
- Výběr Barvy
- Automatizace

Nastavení prostředků (konfigurace)

- Název
- Popis, Info o autorovi
- Umístění v menu
- Verze
- Parametry modulu, jejich rozsahy, defaultní hodnoty
- Podporované formáty
- Režimy zpracování obrazu
- Omezení (velikost obrazu)
- a mnoho dalších

API rozšiřované aplikace

- Strukturální API
- Funkcionální API
- Kombinace obou (volání funkcí, kde parametry jsou struktury)

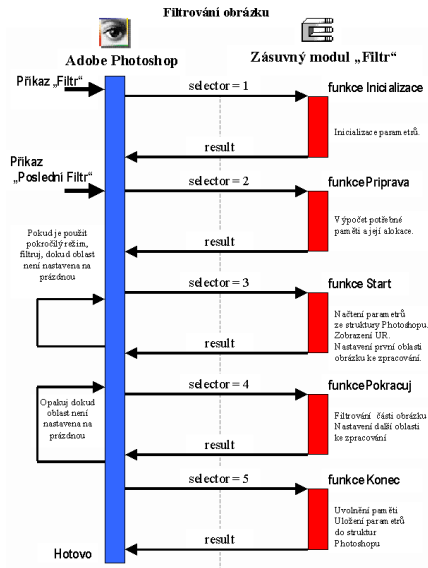
Příklad strukturálního API rozšiřované aplikace

abortProc	This field contains a pointer to the TestAbort call-back function.
progressProc	Function pointer to progress bar callback. Call this during main processing loop. Takes two long-int arguments: current index, and total.
parameters	If your plug-in filter has any parameters that the user can set, you should allocate a relocatable block, store the parameters in the block, and store the block's handle in this field, after your setup dialog returns.
imageSize	This is a Point giving the dimensions of the image or selection.
imageMode	The mode of the image (RGB, CMYK, etc.); see PI-Filter.h for #defines.
filterRect	The overall area (Rect) of the image or bounding box of the selection.

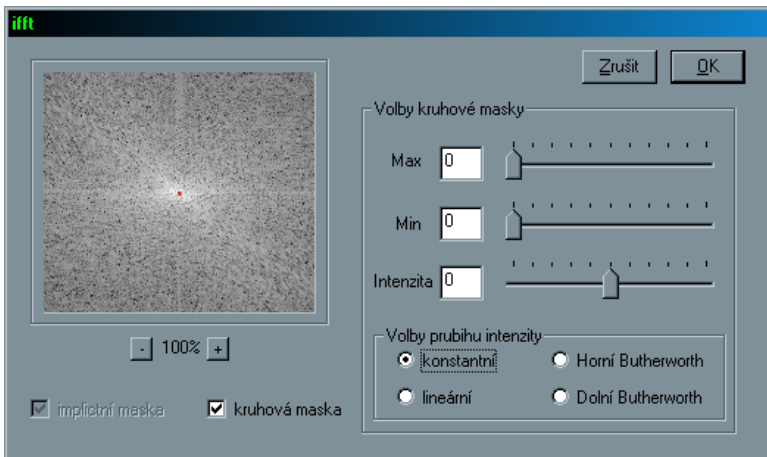
Příklad strukturálního API rozšiřované aplikace

maxSpace	The maximum number of bytes the plug-in can expect to access at once.
bufferSpace	Set this to the number of bytes of memory the plug-in is planning on allocating for internal buffers or tables. Do this prior to the first call for data, in order to minimize scratch disk activity and gain performance.
planes	The total number of planes in the image, including alpha channels.
inLoPlane,	Set these values to tell the host the first and last planes you inHiPlane want to process.
outLoPlane, outHiPlane	Set these to tell the host the first and last output planes you intend to write to. You can write to a subset of inLoPlane/inHiPlane, if necessary. Most times, however, you will set these equal to inLoPlane/inHiPlane.

Životní cyklus modulu



Uživatelské rozhraní modulu



ift

Zrušit OK

Volby kruhové masky

Max 0

Min 0

Intenzita 0

Volby průběhu intenzity

konstantní Horní Butherworth

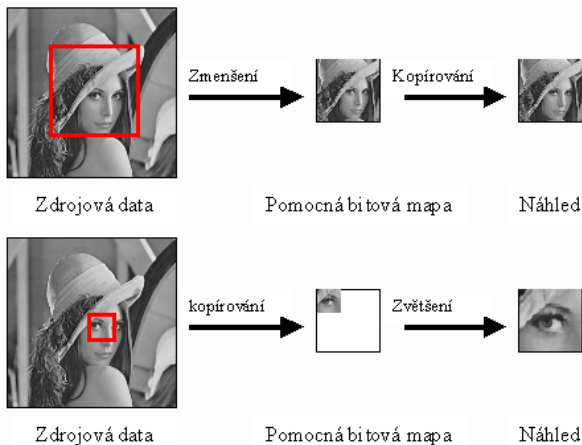
lineární Dolní Butherworth

implicitní maska kruhá maska

- 100% +

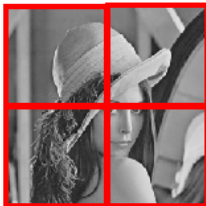
Uživatelské rozhraní modulu (pokr.)

- návrh šablony
- obsluha událostí
- náhled



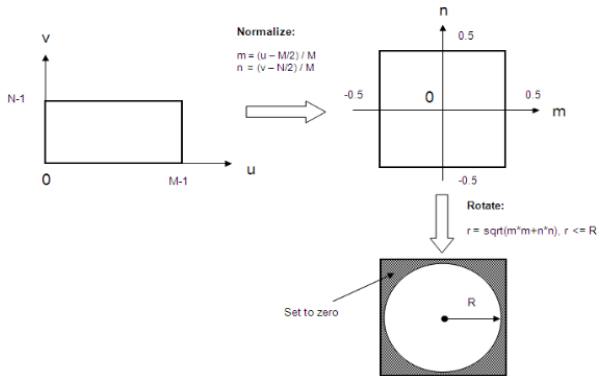
Management paměti

- lze filtrovat po částech?
- bude více průchodů



Použití matematické a grafické funkce

- Vnitřních knihovny.
- Externí knihovny.
- Vlastní implementace - náplň některých přednášek



Co bychom po dnešku měli znát

Témata probraná na dnešní přednášce:

- Základní principy rozšiřování aplikací
- Historie
- Princip zásuvných modulů