

---

# **Rastrové grafické formáty**

**© 1995-2001 Josef Pelikán  
KSVI MFF UK Praha**

e-mail: [Josef.Pelikan@mff.cuni.cz](mailto:Josef.Pelikan@mff.cuni.cz)

WWW: <http://cgg.ms.mff.cuni.cz/~pepca/>

# Grafické formáty

---

## ◆ **rastrové**

- obdélníková **matice pixelů** (“bitmapa”)
- MS-Windows Bitmap (BMP), CompuServe (GIF), Interchange File Format (IFF), JFIF (JPG), PBM/PGM/PPM, Paintbrush (PCX), Macintosh (PICT), Sun Raster Format, Targa (TGA), Tagged Image File Format (TIFF), ...

## ◆ **vektorové**

- posloupnost **objektů** nebo **příkazů** (škálování)
- CorelDraw!™ (CDR), Computer Graphics Metafile (CGM), AutoCAD™ (DXF), HPGL, (Encapsulated) PostScript™, MS-Windows Metafile (WMF), DrawPerfect™ (WPG), ...

# Rastrové grafické formáty

---

- ◆ **formát uložení barev**

- barevná paleta, šedá škála, “true-color”, kanál “ $\alpha$ ”

- ◆ **komprese**

- **bezeztrátová** / **ztrátová**

- **RLE**: PCX, TGA; **LZW**: GIF; **JPEG**: JFIF, TIFF; ..

- ◆ **rozklad obrázku**

- prokládané režimy (GIF, TGA, JPEG, ..)

- ◆ **negrafické informace (TIFF, GIF)**

- ◆ **závislost na HW, přenositelnost (PCX, TGA)**

# PCX (ZSoft Corporation 1988)

---

- ◆ **rastrový formát** (program PC Paintbrush)
- ◆ **omezený počet barev**
  - původně **2 ÷ 16** barev (EGA, VGA)
  - rozšíření: **256** barev, “**true-color**”
- ◆ **HW orientovaný** (IBM CGA, EGA, VGA)
  - prokládané bitové roviny (“color planes”): **RGBI**
- ◆ **jednoduchá RLE komprese**
  - bytově-orientovaná

# Struktura PCX souboru

---



## Hlavička souboru:

- **PCX verze** (2.5, 2.8 bez palety, 2.8 s paletou, 3.0)
- kódování: **RLE**
- počet bitových rovin, počet bitů/pixel/rovinu
- rozměr originální obrazovky
- okno obrázku:  $[X_1, Y_1] - [X_2, Y_2]$
- paleta: **16×3** byty (RGB z  $[0, 255]^3$ ), formát palety: barvy nebo šedé odstíny

# RLE komprese v PCX

---



**N × opakuj 'data'**



**xx ≠ 11 (data < 192)**

**obyčejný datový byte**

**některé soubory se mohou dost prodloužit**

– datový byte > 191 se musí kódovat pomocí “běhu”

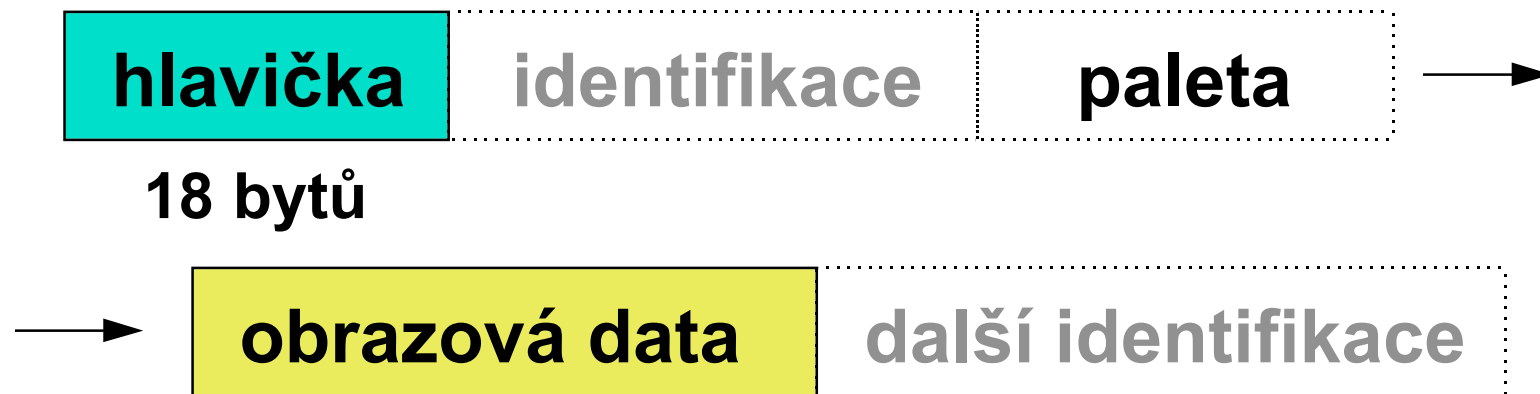
# Targa formát (Truevision Inc.)

---

- ◆ **rastrový formát**
- ◆ **původně HW orientovaný**
  - video-adaptéry **Targa** (Targa 16, Targa 24, ..)
- ◆ **několik různých barevných formátů**
  - **RGB, RGB $\alpha$ , šedé obrázky, obrázky s paletou, atributové bity**
  - různé metody **komprese** (**RLE** komprese je pixelově orientovaná)
- ◆ **různé typy prokládání** (přenos po síti)

# Struktura TGA souboru

---



## Hlavička souboru:

- barevný formát (paleta, RGB, RGB $\alpha$ , šedý obrázek)
- délka identifikace (ASCII text, maximálně 256 znaků)
- typ komprese: bez, RLE, Huffman, delta-modulace
- velikost obrázku: [**X**<sub>0</sub>, **Y**<sub>0</sub>], šířka, výška
- orientace (shora, zdola), typ prokládání (1, 2, 4 fáze)



# Formáty pixelu v TGA

---

paleta,  
šedý obrázek



8 nebo 16 bitů

RGB 16



atribut

16 bitů

RGB 24



24 bitů

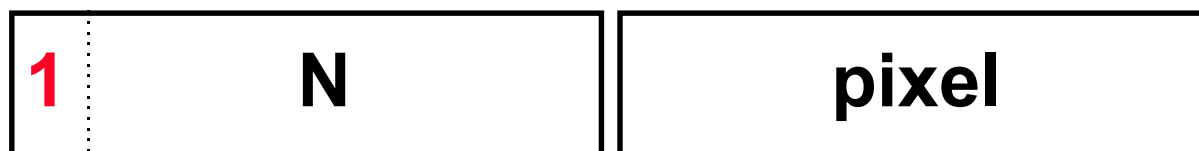
RGB 32



32 bitů

# RLE komprese v TGA

---



**$N+1 \times$  opakuj 'pixel'**



**kopírovací paket**

**$N+1$  pixelů**

**maximální délka paketu je 128 pixelů**

- prodloužení je v nejhorším případě 0.8 % délky souboru

# GIF formát (CompuServe Inc.)

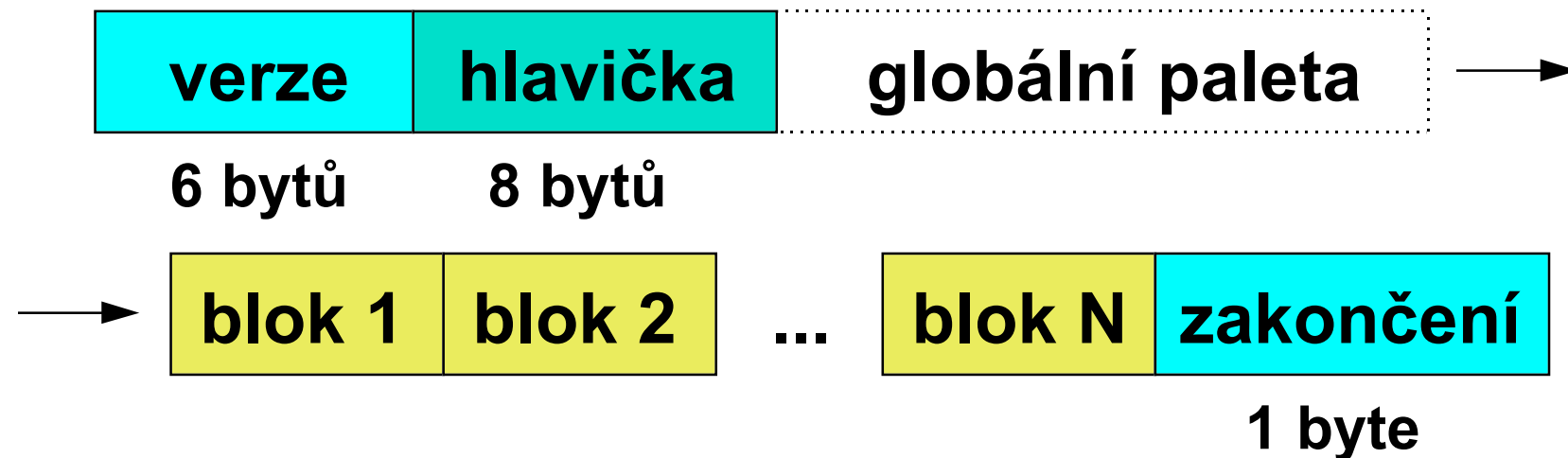
---

## Graphics Interchange Format (verze 87a, 89a)

- ◆ **rastrový formát** relativně **nezávislý na HW**
- ◆ **pouze obrázky s paletou** (max. 256 barev)
- ◆ **LZW komprese** s dynamickou délkou kódu  
– patent UniSys Inc. (licenční poplatky od roku 1995)
- ◆ volitelné 8-fázové **prokládání** (přenos po síti)
- ◆ **další rozšíření**: více obdélníkových obrázků v jednom souboru, definice “průsvitné barvy”, interakce uživatele, výpis textu, aplikační neobrazové informace

# Struktura GIF souboru

---



Verze: **'GIF87a'** nebo **'GIF89a'**

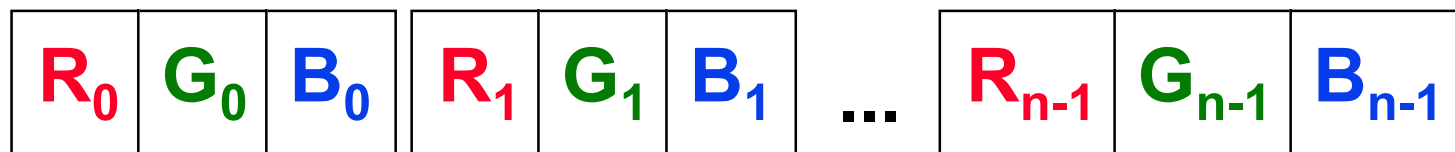
**Globální hlavička:**

- výška a šířka **virtuální obrazovky**, počet bitů na pixel, barva pozadí, “pixel aspect ratio” (4/1 až 1/4)
- **globální paleta**: délka, setřídění (významné barvy jsou na začátku)

# Struktura GIF souboru

---

## Paleta:



(n×3) byty

## Bloky:

- **obrazová** nebo jiná data (poznámky, aplikační data, řídicí bloky)
- **jednotný vnější formát**: starší verze dekodéru může neznámé bloky přeskakovat

# Obrazový blok

---

- ◆ **rozměry výřezu**

- $[X_0, Y_0]$ , šířka, výška

- ◆ **nepovinná lokální paleta**

- počet barev, setřídění (podle důležitosti)?

- ◆ **volba - prokládání**

- 8 fází kreslení obrázku (viz dále)

- ◆ **obrazová data**

- počáteční délka LZW kódu, vlastní kódovaná data

# Prokládání

---

0	I
1	IV
2	III
3	IV
4	II
5	IV
6	III
7	IV
8	I

I fáze:      řádky  $8i$

II fáze:      řádky  $4 + 8i$

III fáze:      řádky  $2 + 4i$

IV fáze:      řádky  $1 + 2i$

# Rozšiřující bloky (verze 89a)

---

## **Blok řízení grafiky:**

- uvolnění grafiky (nechat/smazat/obnovit)
- interakce uživatele, prodleva
- definice transparentní barvy

**Blok komentáře** (jakýkoliv text - pro uživatele)

## **Blok textu:**

- výpis textu na obrazovku (neproporcionální font)

## **Aplikačně závislý blok:**

- libovolná binární data (viz Fractint)



# LZW komprese (Lempel-Ziv-Welch)

---

## ◆ slovníková kompresní metoda

- **slovník**: obsahuje přiřazení “**fráze** → **kód**”
- **fráze**: posloupnost pixelů
- **kód**: **n**-bitové číslo ( $3 \leq n \leq 12$ )

## ➡ v průběhu kódování se mění

- **slovník** (adaptivní přizpůsobení kódovaným datům)
- **délka kódového slova “n”** se zvětšuje po jedné až do 12

# Schema kódovacího algoritmu

---

## ① inicializace

- do slovníku se uloží všechny jednopixelové fráze
- **Act** := “ (prázdný řetězec)

## ② přečti další pixel ze vstupu do **K**

## ③ je fráze **Act + K** uložena ve slovníku?

- **Ano:**     **Act** := **Act + K**
- **Ne:**     zapiš na výstup kód fráze **Act**  
             přidej **Act + K** do slovníku  
             **Act** := **K**

## ④ pokud neskončí vstup, opakuj kroky ② a ③

## ⑤ zapiš na výstup kód fráze **Act**

# Přidávání fráze do slovníku

---

## ➔ počáteční nastavení slovníku:

- kódy  $0 \div 2^p - 1$ : jednopixelové fráze
- kód  $2^p$ : “reset” (inicializace přeplněného slovníku)
- kód  $2^p + 1$ : ukončovací znak (EOF)
- první volný kód fráze:  $2^p + 2$
- počáteční délka kódového slova:  $n = p + 1$  bitů

## ➔ pokud má přidaná fráze kód $2^p$ , zvětším $n$ o 1

- maximální hodnota  $n$  je 12 (4094 fráze)
- při přeplnění zakonzervuji slovník (méně často)  
nebo pošlu “reset” kód (reinicializace slovníku)

# Konec

---

## Další informace:

- **Kay D. C., Levine J. R.: *Graphics file formats*, MGWH, 1994**
- ➔ **WWW:** <http://www.cica.indiana.edu/graphics/>
- ➔ **LAN na Malé Straně:**
  - **barbora\usr:\vyuka\pelikan\zdroj\**
  - **..\formaty\gformats.arj**