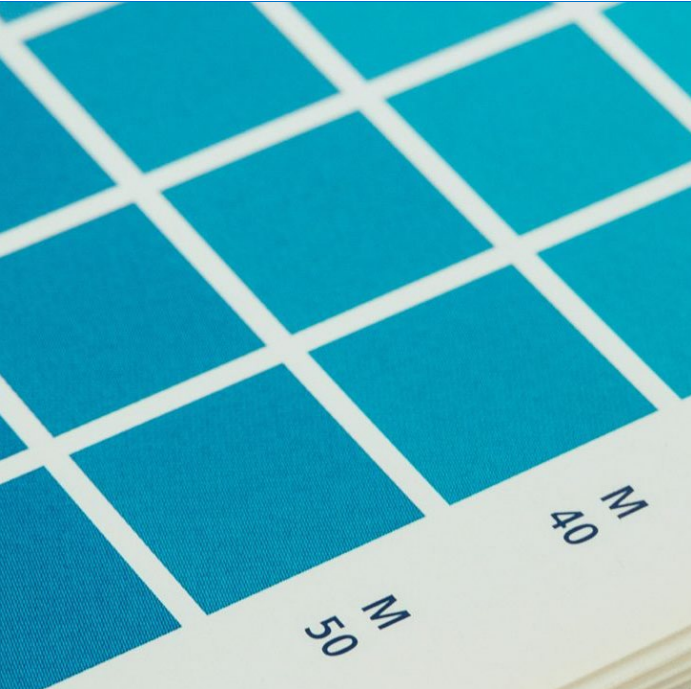




## *Druckvorstufe* 3 — Farbe

Prof. Dr.-Ing. Tenshi Hara  
8. April 2025 | Berlin

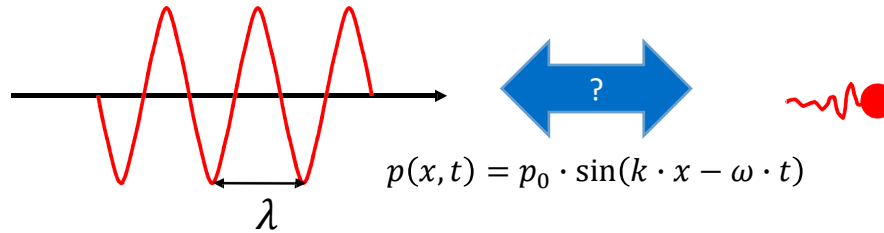


# Licht und Farbmodelle



## Licht – Beschaffenheit

- Licht ist eine in Photonen gequantelte transversale elektromagnetische Welle



- Lichtgeschwindigkeit:  $c = \lambda \cdot f$

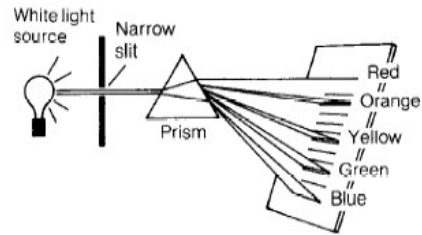
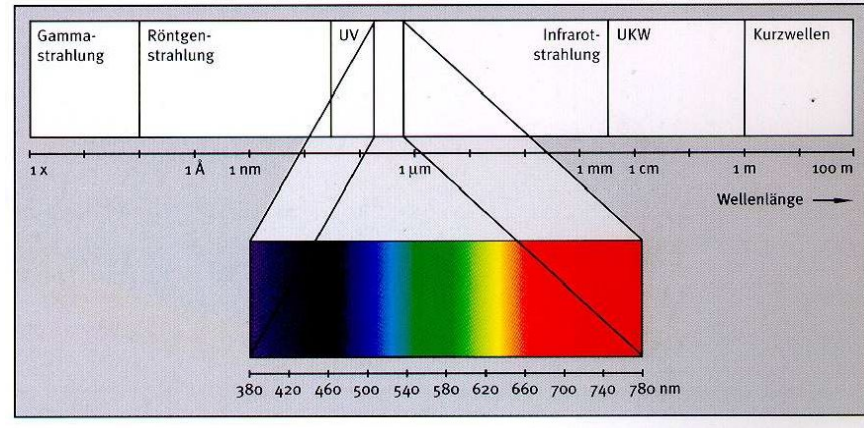
- $c$  nur vom Medium abhängig,

Frequenz ist Maß für Energie:  $E = h \cdot f = \frac{h \cdot c}{\lambda}$

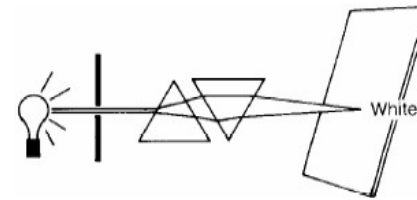
$h = 6,63 \cdot 10^{-34} \text{Js}$  (Planck'sches Wirkungsquantum)



# Licht – Beschaffenheit



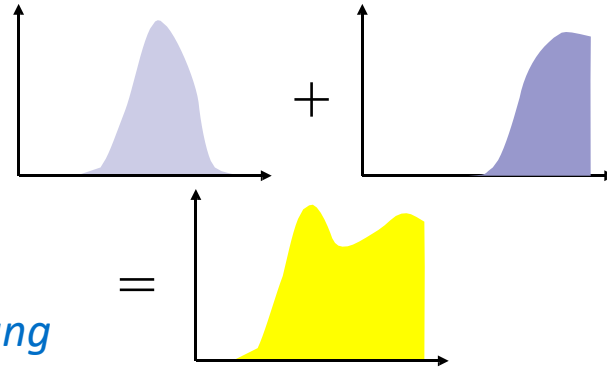
Newton Experiment:  
Aufspaltung



Rekombination



## Licht – Mischung



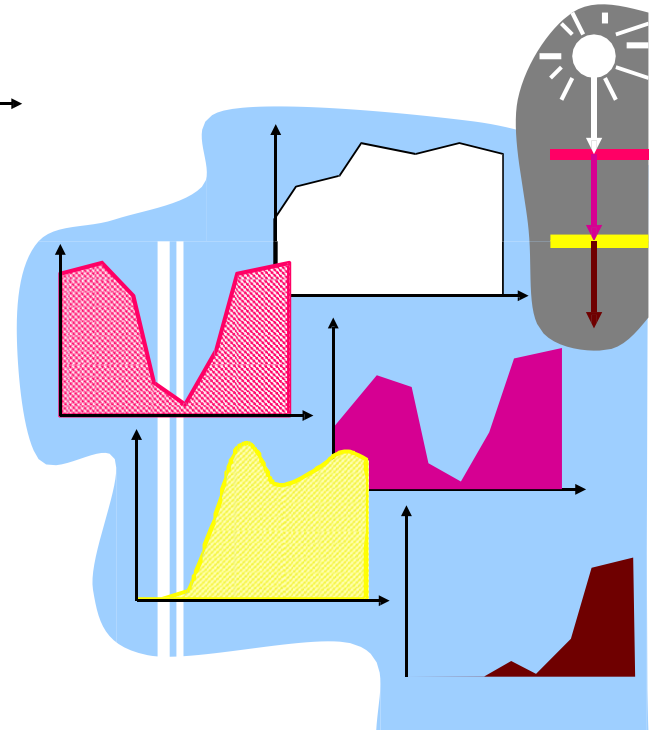
- *additive Farbmischung*

Überlagerung mehrerer Emissionsquellen:  
Spektren addieren sich

- *subtraktive Farbmischung*

Licht fällt auf Absorptions-/Reflektions-Filter:  
Spektren multiplizieren sich unter Filterreduktion

⇒ siehe auch: <https://www.farbe.com/additiv.htm>





## Usability – Farben

- Einsatz von Farben in Text
  - Hervorhebung
  - Bewertung von Worten
- wichtige Grundregeln
  - pro Text soll jede Farbe mit genau einer festen Bedeutung verbunden sein
  - Alltagsbedeutungen von Farben sollten beachtet werden (Kulturkreis beachten, bspw. Europa: rot = Warnung)
  - Farben mit großer spektraler Distanz können nicht gleichzeitig fokussiert werden
  - Farbenblinde sollten inkludiert werden (Simulation möglich)
  - beste Effekte mit sparsam eingesetzten, leuchtenden Farben, die zwischen eher matten Tönen eingesetzt werden





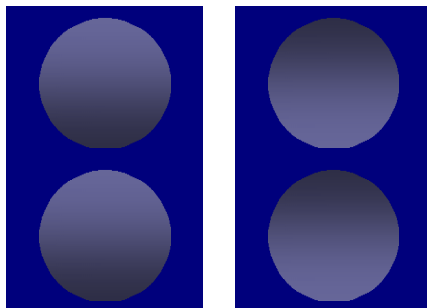
## Perspektive (1/2)



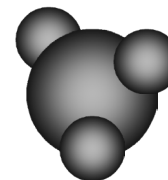
lineare Perspektive



atmosphärische Perspektive



Beleuchtung



Verdeckung



## Perspektive (2/2)



Höhe im Gesichtsfeld



Textur



Beleuchtung



Verdeckung



## Bildschirm-Farbräume



## RGB — Additive Farbmischung

- Primärfarben

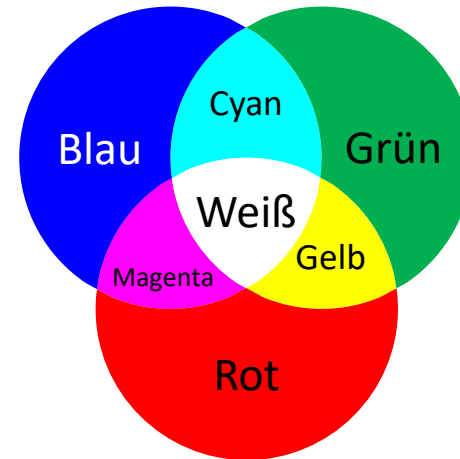
- Rot  $R = (1,0,0)$

- Grün  $G = (0,1,0)$

- Blau  $B = (0,0,1)$

- Mischfarben

$$\forall 0 \leq x \leq 1, x \in \{a, b, c\}: F = aR + bG + cB$$





## RGB-Beispiel

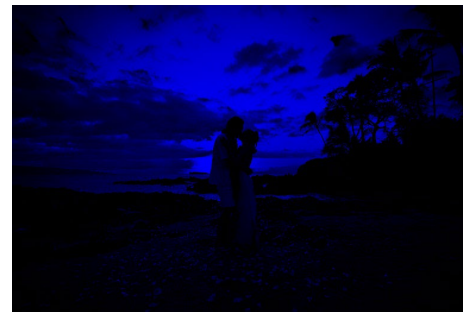
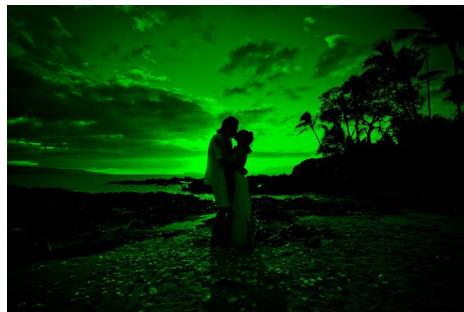
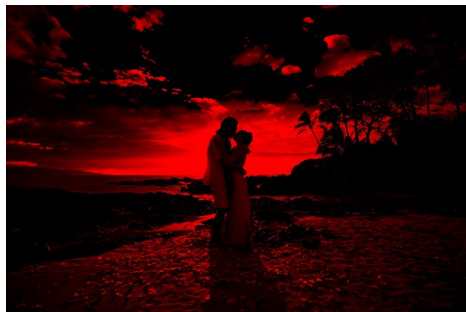


© Karina und Tenshi Hara (23.08.2011)

R

G

B





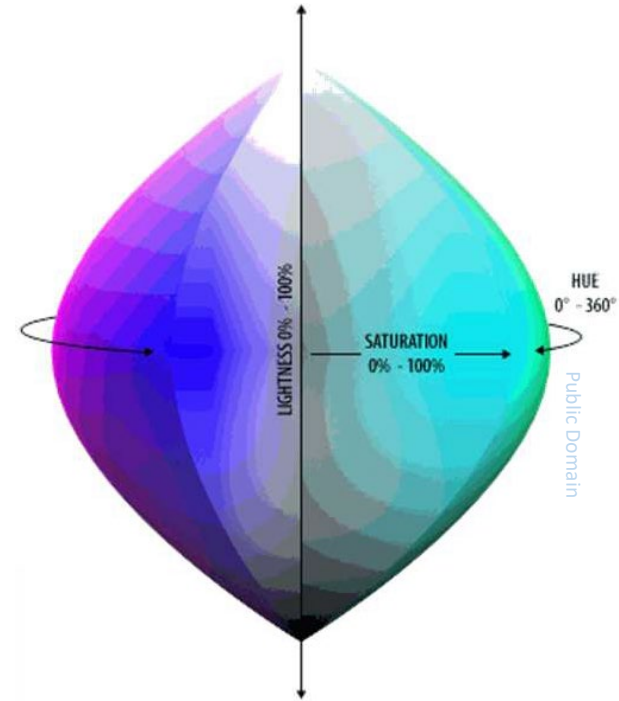
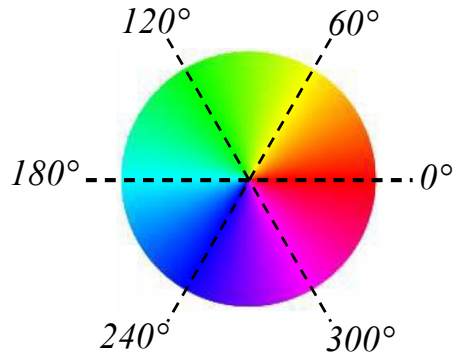
## RGB — True Colour: 16,7 Mio Farben sind wenig!





# HLS

- Hue, Luminance, Saturation
- $H \in [0, 360]$  (0 = 360 = rot)
- $L \in [0, 100]$  (dunkel...hell)
- $S \in [0, 100]$  (matt...brillant)





## Umrechnung RGB $\leftrightarrow$ HLS

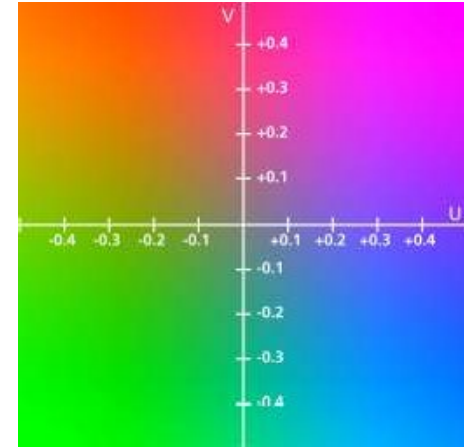
```
void hlsToRgb() {
    H /= 60; L /= 100; S /= 100;
    I = round(H);
    F = H-I;
    if (L <= 0.5) max = L*(1+S)
    else max = L*(1-S)+S;
    min = 2*L - max;
    DM = max - min;
    if (S == 0) [R,G,B] = [L,L,L];
    if (I==0) [R,G,B] = [max, min + F*DM, min]
    if (I==1) [R,G,B] = [min+(1-F)*DM,max,min]
    if (I==2) [R,G,B] = [min,max,min+F*DM]
    if (I==3) [R,G,B] = [min,min+(1-F)*DM,max]
    if (I==4) [R,G,B] = [min+F*DM,min,max]
    if (I==5) [R,G,B] = [max,min,min+(1-F)*DM]
}
```

```
void rgbToHls() {
    max = getMax(R,G,B);
    min = getMin(R,G,B);
    L = (max+min)/2;
    if (min == max) [H,L,S] = [0,L,0];
    else {
        DM = max-min;
        SM = max+min;
        S = (L <= 0.5) ? DM/SM : DM/(2-SM);
        if (R == max) H = (G-B)/DM;
        else
            if (G == max) H = 2+(B-R)/DM;
            else H = 4+(R-G)/DM;
        if (H < 0) H += 6;
    }
    H *= 60;
    L *= 100;
    S *= 100;
}
```



## weitere lineare Farbräume

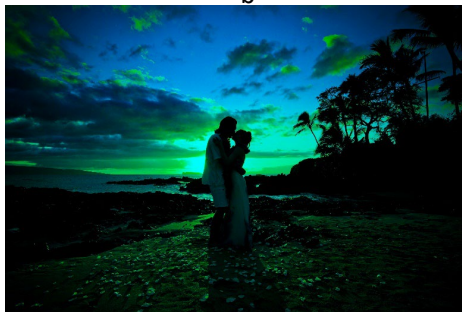
- Lab: physiologisch korrekt und umfassend RGB, CMYK
- YUV:
  - Y ... Helligkeit (Schwarz-Weiß-Kanal)
  - U, V ... Farbbalance
- YC<sub>b</sub>C<sub>r</sub> (digitale Bildformate):



$$f_{YC_bC_r} = \begin{pmatrix} f_Y \\ f_{C_b} \\ f_{C_r} \end{pmatrix} = \begin{pmatrix} 0 \\ 128 \\ 128 \end{pmatrix} + \begin{pmatrix} 0,299 & 0,587 & 0,114 \\ -0,168736 & -0,331264 & 0,5 \\ 0,5 & -0,418688 & -0,081312 \end{pmatrix} \begin{pmatrix} f_R \\ f_G \\ f_B \end{pmatrix}$$

© Karina und Tenshi Hara (23.08.2011)

Y

 $C_b$  $\dot{C}_r$ 



## Druck-Farbräume



## Druckfarbräume

- Pantone (pantone.com)
  - Farbstandard für über 3000 Volltonfarben
  - Vielseitigkeit für Graphiker
  - schlechte Mischbarkeit (Farbseparation)
  - Definition über Farbmuster
- Truematch (truematch.com)
  - System zum Farbabgleich
  - CMYK-Farben aus >2000 Prozessfarben
- Focoltone (focoltone.com)
  - kalkulierbare Prozessfarben
  - 763 Farben aus CMYK

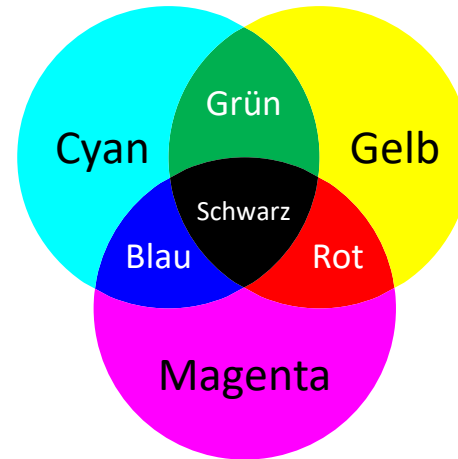




## CMY — Subtraktive Farbmischung

- CMY-Farbmodell eignet sich für die Körperfarbmischung (d.h. Lasur- und Partikelfarben)
- Umrechnung vom RGB-Modell ins CMY-Modell:
- $(r, g, b) = (1, 1, 1) - (c, m, y)$

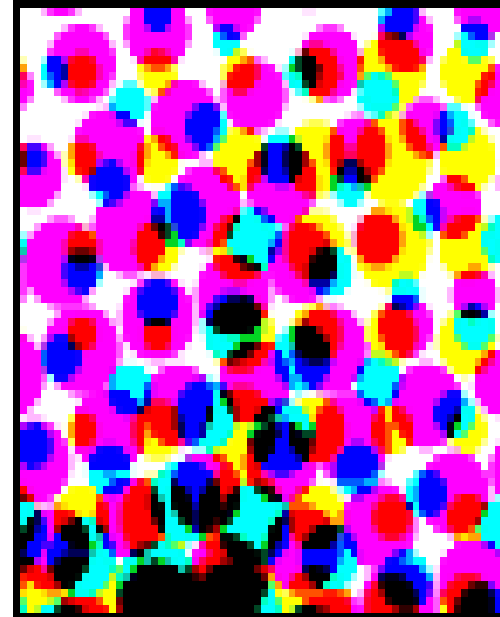
Darstellung für Drucker macht nur bei gleichem Farbgamut Sinn!





## CMYK — Prozess-Farben-Modell

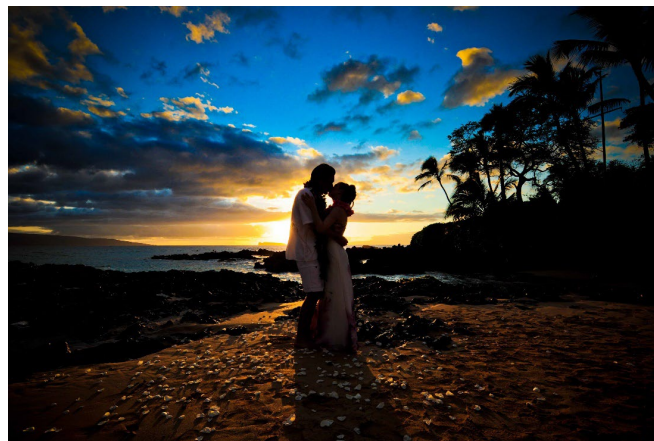
- Key-Kanal für Schwarzanteile
- Schwarz braucht dann nicht mit Buntfarben gemischt werden  
→ Ausnahme: **fettes Schwarz**



Public Domain



# CMYK-Beispiel



© Karina und Tenshi Hara (23.08.2011)

C

M

Y

K



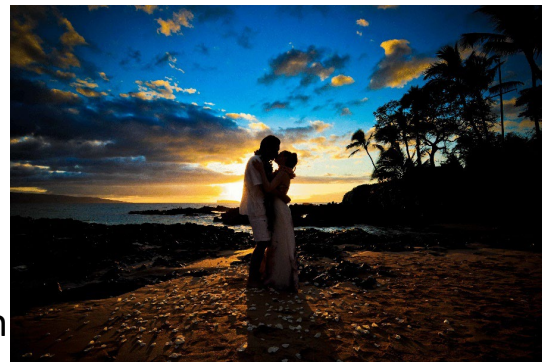


## Farbreduktion für den Druck



24 Bit - True Colour

© Karina und Tenshi Hara (23.08.2011)



8 Bit – 256 Farben



4 Bit – 16 Farben



3 Bit – 8 Farben



## Bildanpassung für den Druck

$$f'(x,y) = \frac{1}{9} \sum_{p=x-1}^{x+1} \sum_{q=y-1}^{y+1} f(p,q)$$

$$f'(x,y) = \begin{pmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{pmatrix} * \begin{pmatrix} f(x-1, y-1) & f(x, y-1) & f(x+1, y-1) \\ f(x-1, y) & f(x, y) & f(x+1, y) \\ f(x-1, y+1) & f(x, y+1) & f(x+1, y+1) \end{pmatrix}$$



Original



Weichzeichner-  
Filter



Scharfzeichner-  
Filter

Public Domain



## geometrische Operationen

- geometrische Operationen wie Verkleinerung/Vergrößerung verändern die Auflösung und führen bei Rastermustern im Bild zu Moiré-Effekten in der Verkleinerung/Vergrößerung (Original → a)
- Vermeidung durch Weichzeichner (Original → Filter → b)

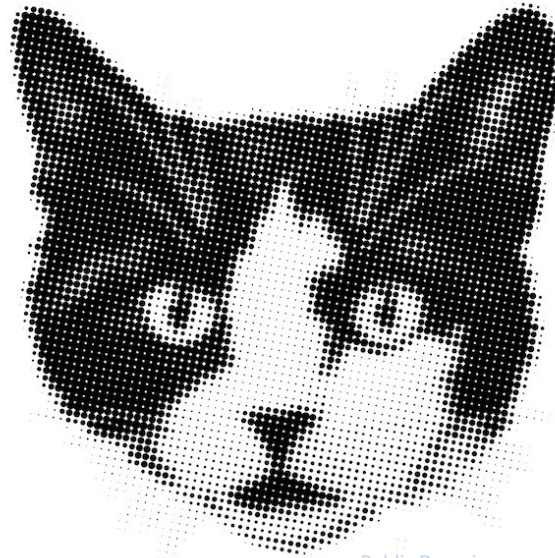
Original





## Druckraster

- sparsame Alternative zur Farbmischung
- basiert auf Wahrnehmungseigenschaften (siehe Gestaltgesetze)

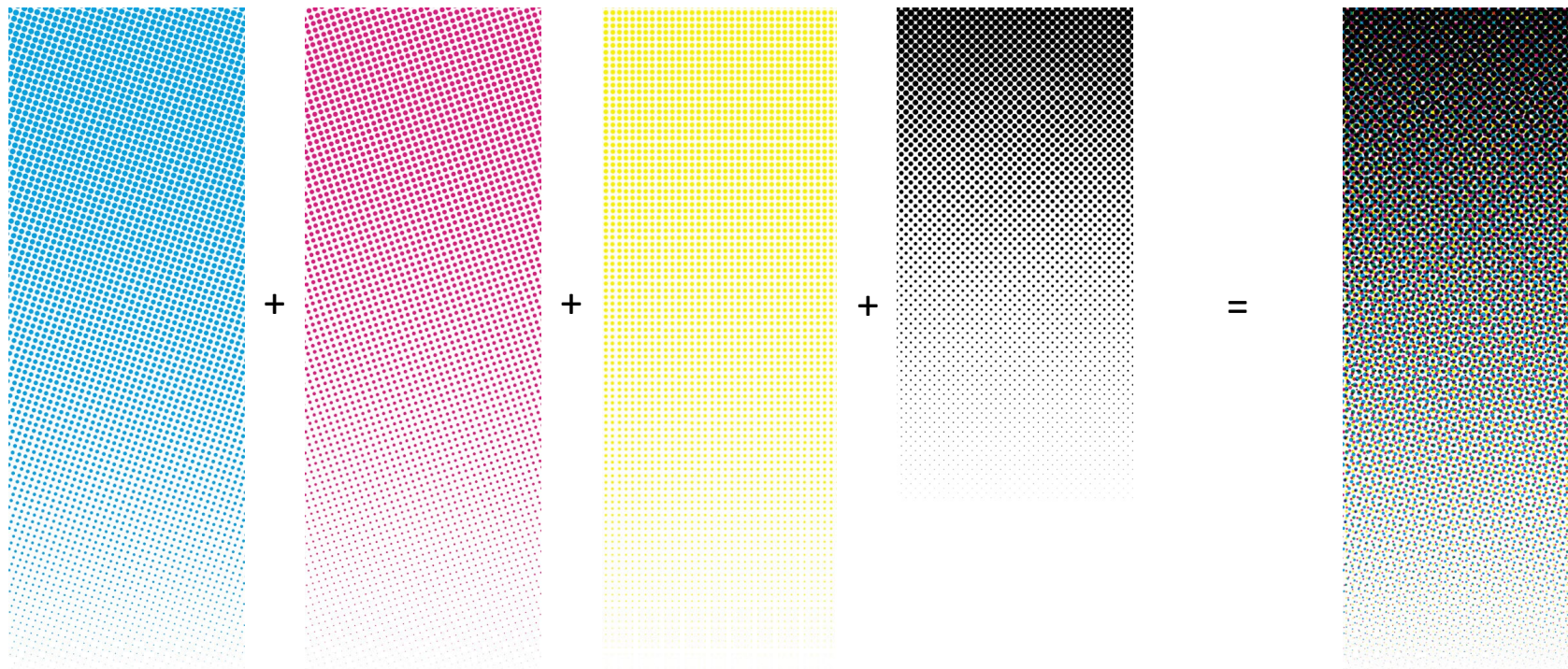


Public Domain



# Rasterprobleme

Vorsicht bei der „Mischung“ von Rastern!





## Wichtige Bildaustauschformate



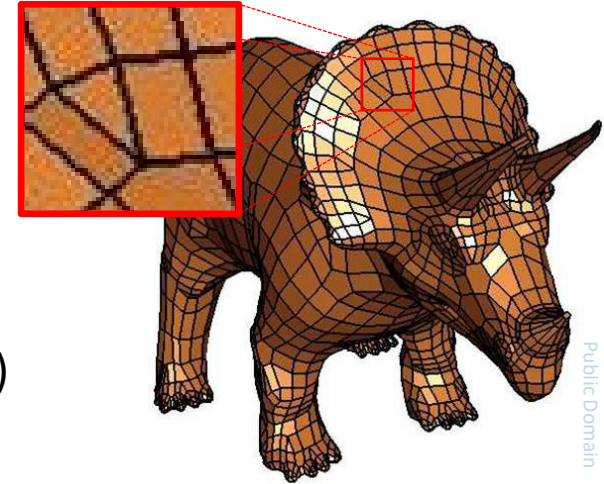
## Joint Photographic Experts Group (JPEG)

- Basisformat für komprimierte Bilder in PDF
- Kompression durch verlustbehaftete Verarbeitungsschritte in Zusammenwirken mit Huffman-Kodierung
- Blockbildung, stufige Übergänge, Farbeffekte an Graukeilen
- Nachfolger JPEG-2000 ist wesentlich weniger anfällig für diese Art von Artefakten und kann ICC-Profile einbetten



## Einsatz von JPEG

- geeignet für Fotos
- ungeeignet für
  - Strichzeichnungen
  - Schwarzweißbilder  
(nur Schwarz und Weiß, d.h. 1 Bit pro Bildpunkt)
  - gerasterte Bilder (bspw. im Zeitungsdruck)



Public Domain



## JPEG 2000, JP2

- verlustfreie oder verlustbehaftete Komprimierung
- Format basiert auf XML
- kann Metadaten enthalten, die das Verwalten und Auffinden erleichtern
- bestimmte Bildregionen können in höherer Qualität gespeichert werden (geringere Kompression oder verlustfrei)
- Interlacing ähnlich PNG  
(für Druck unwichtig, außer bei Layer-Druck-Verfahren)



## Tagged Image File (TIFF)

- Vorteile
  - unterstützt viele Farbräume u.a. RGB und LAB
  - kann ICC-Profile einbetten
  - verlustfreie Komprimierung möglich mit LZ77- oder LZW-Algorithmus
- Nachteil
  - Komprimierungsrate ist oft zu gering  
(bspw. DIN-A4-Seite mit 600dpi und 256 Farben: TIF 32 MB, PNG 266 KB)



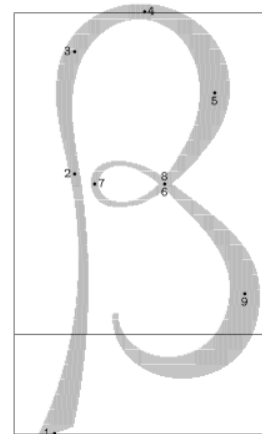
## Adobe Photoshop Document (PSD)

- Vorteile
  - unterstützt getrennte Farbkanäle (auch Alpha) und Ebenen
  - speichert Bearbeitungen, ICC-Profile
  - Text wird als Objekt im Dokument gespeichert, erst auf Ausgabegerät gerastert
  - verlustfreie Weiterbearbeitung möglich
  - de facto Standard in der Druckindustrie (neben PDF)
- Nachteile
  - proprietäres Bildformat
  - kann nicht von allen Programmen gelesen werden



# Vektorschriften

- True-Type Fonts
  - Glyphen durch Befüllen ihrer Kontur, zusammengesetzt aus Bézier-Kurven 1. bis 3. Grades
  - die interpolierten Anfangs- und Endpunkte nennt man „on points“, die Zwischenpunkte „off points“
- Meta Font (bspw. mit  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ )
  - Glyphen definiert durch interpolierende Kurve eines Pinsels mit wählbarem Profil
  - Profil kann entlang der Kurve Größe und Orientierung ändern
  - sehr mächtig: kann programmiert werden und Gleichungssysteme lösen



```

u#:=4/9pt#;
define_pixels(u);
beginchar(66,13u#,16u#,5u#);"Letter beta";
x1=2u; x2=x3=3u;
bot y1=-5u;
y2=8u; y3=14u; x4=6.5u;
top y4=h;
z5=(10u,12u); z6=(7.5u,7.5u); z8=z6;
z7=(4u,7.5u); z9=(11.5u,2u); z0=(5u,u);
penpos1(2u,20);
penpos2(.5u,0);
penpos3(u,-45);
penpos4(.8u,-90);
penpos5(1.5u,-180);
penpos6(.4u,150);
penpos7(.4u,0);
penpos8(.4u,210);
penpos9(1.5u,-180);
penpos0(.3u,20);
pickup pencircle;
penstroke z1e..z2e..z3e..z4e..z5e..z6e..
{up}z7e..z8e..z9e..{up}z0e;
labels(range 1 thru 9);
endchar;
end

```