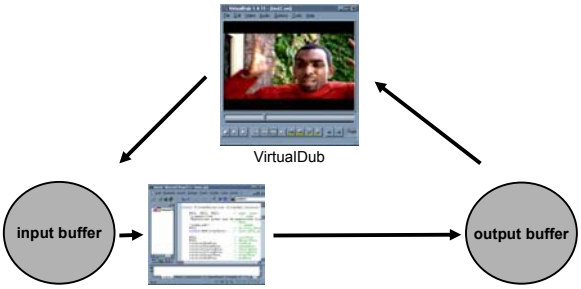
Ulrike Schaal - David Eißler - Marc-Oliver Pahl

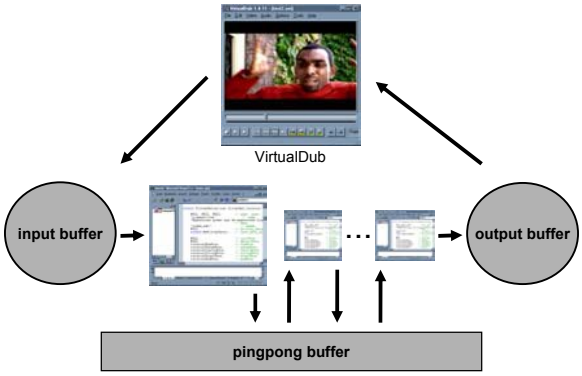
Filterprogrammierung in Virtual Dub 1.4b

Wird sich mit *VirtualDub 2.0* ändern...

Filterkette



Filterkette



Probleme

- Alle Buffer in **RGB32**
- kein Zugriff auf **nachfolgende Bilder**
- **Anzahl** und **Geschwindigkeit** der Bilder kann nicht geändert werden
  - keine Zeitlupe, kein Auslassen von Bildern

Zustände

buffer negotiation
Eingefügte Filter werden nacheinander abgefragt,
damit die benötigten Buffer (Größe) erstellt werden können.

filter initialization
Die Initialisierungsmethode des Filters wird aufgerufen.
(Eigene Buffer etc. erstellen)

running
Für jedes Frame werden alle Filter nacheinander abgearbeitet.

filter shutdown
Die DeInitialisierungsmethode des Filters wird aufgerufen.
(Speicher aufräumen)

Progammierung

Für einen lauffähigen Filter...

...muss man implementieren: **runProc**

...kann man implementieren:

initProc	stringProc
deinitProc	startProc
paramProc	endProc
configProc	

runProc-Gerüst

```
int tutorialRunProc(const FilterActivation *fa, const FilterFunctions *ff) {
        PixDim w, h;
        Pixel32 *src, *dst;

        src = (Pixel32 *)fa->src.data;
        dst = (Pixel32 *)fa->dst.data;

        h = fa->src.h;
                do {
                        w = fa->src.w;

                        do {
                                ++src, ++dst;
                        } while(--w);

                src = (Pixel32 *)((char *)src + fa->src.modulo);
                dst = (Pixel32 *)((char *)dst + fa->dst.modulo);
                } while(--h);
        return 0;
}
```


unsere runProc

```
Pixel32 old_pixel, new_pixel;

old_pixel = *src++;

if (anz<lines) {
        *dst++ = old_pixel;
} else {
        // red = (old_pixel & 0xFF0000)
        // green = (old_pixel>>8) & 0xff
        // blue = old_pixel & 0xff
        new_pixel =
                ((unsigned long)((old_pixel & 0xFF0000) * alpha) & 0xFF0000)
              + ((unsigned long)((old_pixel & 0x00FF00) * alpha) & 0x00FF00)
              + (unsigned long)((old_pixel & 0x0000FF) * alpha);

                *dst++ = new_pixel;

        }
```


unser Filter