

Dokumentation

component^{version three}**library** für Adobe Illustrator
und
component^{version three}**library** **SA** (StandAlone)

Alexander Franzen
Nov 2006

Zum Copyright

Für die **componentlibrary** in ihrer Gesamtheit wird von mir kein Copyright beansprucht. Dies bedeutet insbesondere, dass die **componentlibrary** frei verteilt und kopiert werden darf. Darüber hinaus ist die Benutzung der **componentlibrary** und der mit ihrer Hilfe erstellten Illustrationen für alle privaten und wissenschaftlichen nichtkommerziellen Zwecke ausdrücklich gestattet.

Unbeeinträchtigt davon bleibt allerdings das Urheberrecht. Dies bedeutet insbesondere, dass die **componentlibrary** nicht kommerziell vermarktet oder in veränderter Form weiterverbreitet werden darf.

Weiterhin gilt: Wird ein Gesamtwerk (etwa eine Diplom- oder Doktorarbeit) in wesentlichen Teilen mit Hilfe der **componentlibrary** illustriert, beanspruche ich einen Vermerk auf die **componentlibrary** und deren Urheber – nämlich mich. Hierzu kann beispielsweise das „Credit Logo“ dienen.

Was ist die ComponentLibrary?

Bei der **componentlibrary** handelt es sich um eine Sammlung von Icons, die optische Komponenten darstellen. Ursprünglich für eine Diplomarbeit entwickelt, dienen diese Icons dazu, optische Versuchsaufbauten illustrativ darzustellen. Der Anspruch bei der Gestaltung der Icons bestand einerseits darin, den größtmöglichen Wiedererkennungswert für jede Komponente zu erreichen. Damit ist der Grundstein für eine gute *Lesbarkeit* der gesamten Illustration gelegt. Weiterhin sollte eine kontrastreiche Darstellung – sowohl bei der Bildschirmdarstellung, als auch bei gedruckten Abbildungen – erreicht werden. Die einzelnen Komponenten sind – auch im Hinterkopf behaltend, welche Komponenten häufig gemeinsam auftreten – aufeinander abgestimmt, um auch in eng bevölkerten Regionen einer Abbildung Abgrenzung und Struktur zu erhalten. Außerdem sind die Komponenten so gestaltet, dass die Abbildungen über einen möglichst großen Bereich der Reproduktionsskalierung *funktionieren*: Sowohl stark verkleinerte Abbildungen in einer wissenschaftlichen Publikation, als auch groß gezogene Illustrationen im Poster-Format zeigen größtmögliche Lesbarkeit und Strukturierung.

Insgesamt standen bei der Entwicklung der **componentlibrary** also dieselben Anforderungen im Vordergrund, wie sie sich in der Typographie finden. Die ästhetische Gestaltung tritt dabei hinter Aspekten der Ergonomie und *Lesbarkeit* zurück. Die Zitate *Gute Typographie bemerkt man nicht* und *Typographie ist Höflichkeit gegenüber dem Leser* sollten auch auf die **componentlibrary** übertragbar sein.

Die ersten Versionen einzelner Icons sind als Vektorgrafiken in Freehand entstanden. Eine parallel entwickelte Version mit reduziertem Kontrastumfang für die Bildschirmdarstellung hat sich nicht durchgesetzt und ist mittlerweile ausgestorben. Um die Handhabung der **componentlibrary** zu verbessern, ist im dritten Update-Durchgang eine Symbol-Bibliothek für Adobe Illustrator 10, kurz darauf für Adobe Illustrator CS entstanden. Die Verwendung dieser Variante ist am praktischsten und wird empfohlen. Um die **componentlibrary** auch Menschen zugänglich zu machen, die den Adobe Illustrator nicht verwenden wollen oder können, gibt es nun auch eine **componentlibrary SA** (StandAlone) Version. Damit kommen nun auch Linux-User in den Genuß.

Kombatibilität

Die **componentlibrary** benötigt Adobe Illustrator CS (manchmal als Adobe Illustrator 11 bezeichnet) oder Adobe Illustrator CS2. Die Verwendung mit älteren Versionen funktioniert eingeschränkt unter Mac OSX, wird aber ausdrücklich nicht empfohlen. Muss man aus technischen Gründen mit einer älteren Version von Adobe Illustrator arbeiten, so empfiehlt sich die Verwendung der **componentlibrary SA** StandAlone Version.

CompLib für Adobe Illustrator

Die **componentlibrary** für Adobe Illustrator liegt als einzelne .ai-Datei vor. Dabei handelt es sich allerdings nicht um eine gewöhnliche Adobe Illustrator datei, die schlicht geöffnet werden kann, sondern um eine Symbol-Bibliothek. Zur Installation muss dieses file an eine bestimmte Stelle der Illustrator Dateistruktur verschoben werden, nämlich:

Adobe Illustrator CS/Vorgaben/Symbole/

Nach einem Neustart von Illustrator kann nun per

Window>Symbol Libraries>ComponentLibrary

eine Symbolpalette geöffnet werden, aus der die einzelnen Komponenten einfach per *drag and drop* in die Grafik hineingezogen werden können. Die entstehende Grafik ist nicht von der **componentlibrary** abhängig. Das heißt, ein mit der **componentlibrary** erzeugtes file kann auch auf einem Rechner geöffnet, betrachtet und verändert werden, auf dem die **componentlibrary** nicht vorhanden ist.

CompLib StandAlone (SA)

Mit der StandAlone Version der **componentlibrary** ist es möglich *ohne* Verwendung von Adobe Illustrator Abbildungen von optischen Aufbauten zu erstellen. Damit kommen etwa Linux-User zum Zuge (für dieses Betriebssystem existiert der Adobe Illustrator nicht). Außerdem ist diese Version der ComponentLibrary etwa geeignet um auf die Schnelle eine kleine Abbildung etwa für eine Power Point- oder Keynote-Präsentation zusammen zu stellen.

Bei der StandAlone-Version der **componentlibrary** handelt es sich schlicht um eine Ansammlung von Bildern im .png-Format in reprofähiger Auflösung. Damit ist keine Installation notwendig. Abhängig vom verwendeten Layoutprogramm werden die einzelnen Komponenten nach Bedarf per *Import...* oder *Place...* in die Grafik geladen und dort positioniert. Um Problemen mit überlappenden Strahlengängen u. ä. aus dem Weg zu gehen, wurden in die Bilder sorgfältig Transparenzmasken eingearbeitet. Die Resultate lassen sich nicht von einer mit der *echten componentlibrary* erstellten Illustration unterscheiden.

Application Notes

Aufbau einer Illustration

Teilt man eine Illustration durch zwei horizontale und zwei vertikale Linien in neun Teile, so erfolgt aus physiologischen Gründen der Einstieg in die Illustration im Bereich der rechten unteren Ecke des linken obersten Teils. Dies sollte man bei der Konzeptionierung einer Illustration berücksichtigen, indem man diese derart aufbaut, dass an dieser Stelle auch der logische Einstieg in den dargestellten Versuchsaufbau möglich ist. Weiterhin wird ein logischer Zusammenhang leichter verständlich, wenn dieser von links nach rechts und von oben nach unten (normale Leserichtung) gelesen werden kann.

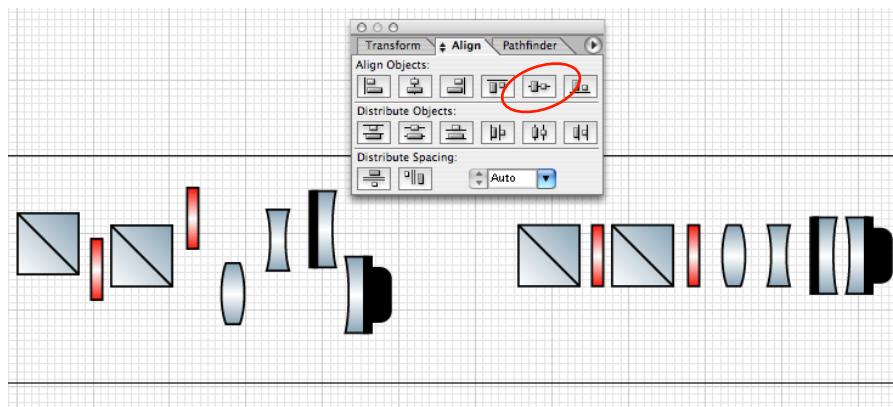
Technisches

- Alle Abbildungen, die Komponenten der **componentlibrary** enthalten, sollten im RGB-Farbraum angelegt werden. Dem Autor fällt – nebenbei bemerkt – auch gar keine Anwendung im Einzugsbereich der ComponentLibrary ein, für die der CMYK-Farbraum sinnvoll wäre. . .
- Illustrationen sollten grundsätzlich in 1:1 angelegt werden – d.h. ohne Skalierung der optischen Komponenten. Eine Anpassung an das Reproformat sollte erst dann erfolgen, wenn die gesamte Abbildung fertiggestellt ist.

- Um die Arbeit zu vereinfachen, sollte auf einem magnetischen Raster gearbeitet werden. Derartiges wird von fast allen gängigen Layout-Programmen unterstützt. Die Ausrichtbarkeit von Komponenten ist gewährleistet, wenn die Rasterweite 1 mm beträgt.
- Man sollte generell der Versuchung widerstehen, an den Komponenten skalierend herumzuzerren. Manchmal ist es allerdings sinnvoll, elektronische Komponenten zu verkleinern, damit sie visuell hinter den optischen Aufbau zurücktreten. In diesem Fall empfiehlt sich eine Skalierung mit dem Faktor 0,5. Dann sollte allerdings auch die Rasterweite des zugrunde liegenden Rasters auf 0,5 mm verkleinert werden.

Hilfreiche Hinweise

Ausrichtungen. Von vielen unbeachtet sind die hilfreichen Ausricht-Funktionen, die Adobe Illustrator (und andere Layout-Programme) an Bord haben. Mit diesen sollte man sich allerdings einmal auseinandersetzen. Beispielsweise erfordert es nur einen einzigen Mausklick, siebzehn Komponenten entlang eines geraden Laserstrahls so auszurichten, dass diese mittig getroffen werden.



Overlays. Ähnliches gilt für die Ebenen-Funktionen von Adobe Illustrator. Ein gutes Vorgehen ist es, auf einem obersten Layer Beschriftungen zu erstellen, darunter die optischen Komponenten, darunter den Strahlengang, darunter Elektronik und Verkabelung. Geht man so vor, ist es ein Leichtes, mit einem einzigen Mausklick verschiedene Versionen derselben Abbildung zu erstellen – zum Beispiel einmal mit und einmal ohne Elektronik. Dies ist etwa zur Erläuterung während eines Vortrages mehr als hilfreich.

Taucht eine Komponente in einer Illustration mehrfach auf, ist es nicht nötig, jede Kopie aus der Symbol-Palette per drag and drop heraus zu ziehen. Eine einmal platzierte Komponente kann auch per *kopieren und einfügen* oder *duplizieren* vervielfältigt werden.

Feedback

Für Feedback in jedweder Form bin ich über alle Maßen dankbar. Insbesondere gilt dies für Hinweise auf fehlende oder unzureichende Komponenten. Werden weitere Komponenten benötigt bitte ich um Zusendung einer Skizze, die ich gerne für die **componentlibrary** umsetze und beim nächsten Update berücksichtige.

Reference Sheets

Auf den folgenden Seiten findet sich eine Übersicht über die Komponenten, die in der **componentlibrary** enthalten sind. Diese Komponenten zerfallen in drei Gruppen: *BasicComponents*, *ElectronicComponents* und *CompoundComponents*. In blau ist jeweils der Dateiname aus der ComponentLibrary SA angegeben.



Beamsplitter (Cube)

[b-bsp.png](#)
[b-bsp.svg](#)



Simple mirror

[b-mir.png](#)
[b-mir.svg](#)



Simple mirror w/ PZT

[b-mirpzt.png](#)
[b-mirpzt.svg](#)



Dichroic mirror (red/grn)

[b-dicred.png](#); [b-dicgrn.png](#)
[b-dicred.svg](#); [b-dicgrn.svg](#)



Waveplates

[b-wpyel.png](#); [b-wpred.png](#); [b-wpgn.png](#)
[b-wpyel.svg](#); [b-wpred.svg](#); [b-wpgn.svg](#)



Dichroic cube

[b-diccube.png](#)
[b-diccube.svg](#)



Crystal flat/curved

[b-crystalfc.png](#)
[b-crystalfc.svg](#)



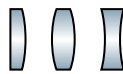
Crystal curved/curved

[b-crystalcc.png](#)
[b-crystalcc.svg](#)



Crystal flat/flat

[b-crystalff.png](#)
[b-crystalff.svg](#)



Lens

[b-lens1.png](#); [b-lens2.png](#); [b-lens3.png](#)
[b-lens1.svg](#); [b-lens2.svg](#); [b-lens3.svg](#)



Beamsplitter (plate)

[b-bspcube.png](#)
[b-bspcube.svg](#)



Curved mirror

[b-mirc.png](#)
[b-mirc.svg](#)



Curved mirror w/ PZT

[b-mircpzt.png](#)
[b-mircpzt.svg](#)



Simple grating

[b-grat.png](#)
[b-grat.svg](#)



Beam dump

[b-dump.png](#)
[b-dump.svg](#)



Fiber Coupler

[b-coupler.png](#)
[b-coupler.svg](#)



NPRO

[b-npro.png](#)
[b-npro.svg](#)



Optical phase shifter

[b-phase.png](#)
[b-phase.svg](#)



Photodetector 1 & 2

[e-pd1.png](#); [e-pd2.png](#)

[e-pd1.svg](#); [e-pd2.svg](#)



Photodetector green 1 & 2

[e-pdgrn1.png](#); [e-pdgrn2.png](#)

[e-pdgrn1.svg](#); [e-pdgrn2.svg](#)



Mixer

[e-mix.png](#)

[e-mix.svg](#)



Frequency generator 1

[e-frq1.png](#)

[e-frq1.svg](#)



Frequency generator 1

[e-frq2.png](#)

[e-frq2.svg](#)



Sum

[e-sum.png](#)

[e-sum.svg](#)



Difference

[e-diff.png](#)

[e-diff.svg](#)



Sum/Difference

[e-sumdiff.png](#)

[e-sumdiff.svg](#)



Highpass filter

[e-hipass.png](#)

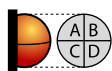
[e-hipass.svg](#)



Lowpass filter

[e-lopas.png](#)

[e-lopas.svg](#)



Quadrant photodetector

[e-qpd.png](#)

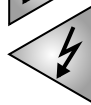
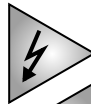
[e-qpd.svg](#)



Spectrum analyzer

[e-spekki.png](#)

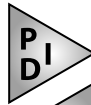
[e-spekki.svg](#)



High voltage amplifier

[e-hvampright.png](#); [e-hvamplleft.png](#)

[e-hvampright.svg](#); [e-hvamplleft.svg](#)



Servo

[e-servoright.png](#); [e-servoleft.png](#)

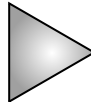
[e-servoright.svg](#); [e-servoleft.svg](#)



Computer

[e-computer.png](#)

[e-computer.svg](#)



Generic Amplifier

[e-amp.png](#)

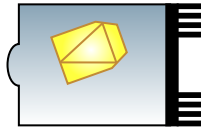
[e-amp.svg](#)



Laser variant 1

[c-laser1.png](#)

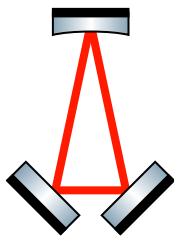
[c-laser1.svg](#)



Laser variant 2

[c-laser2.png](#)

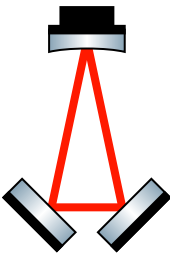
[c-laser2.svg](#)



Ring mode cleaner

[c-modeclean.png](#)

[c-modeclean.svg](#)



Ring mode cleaner w/ PZT

[c-modecleanpzt.png](#)

[c-modecleanpzt.svg](#)



AOM

[c-aom.png](#)

[c-aom.svg](#)



EOM variant 1

[c-eom1.png](#)

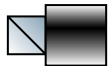
[c-eom1.svg](#)



EOM variant 2

[c-eom2.png](#)

[c-eom2.svg](#)



Rotator

[c-rotator.png](#)

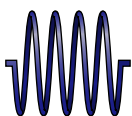
[c-rotator.svg](#)



Isolator

[c-isolator.png](#)

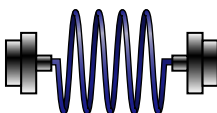
[c-isolator.svg](#)



Fiber

[c-fiber.png](#)

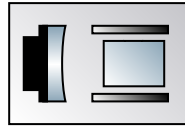
[c-fiber.svg](#)



Fiber w/ couplers

[c-fibercoupl.png](#)

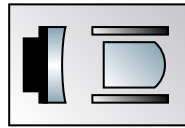
[c-fibercoupl.svg](#)



Nonlinear cavity flat/flat w/ plates

[c-opaffplates.png](#)

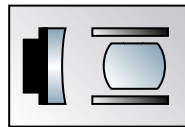
[c-opaffplates.svg](#)



Nonlinear cavity flat/curved w/ plates

[e-c-opafcplates.png](#)

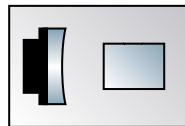
[e-c-opafcplates.svg](#)



Nonlinear cavity curved/curved w/ plates

[c-opaccplates.png](#)

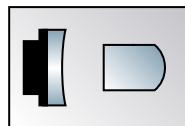
[c-opaccplates.svg](#)



Nonlinear cavity flat/flat

[c-opaff.png](#)

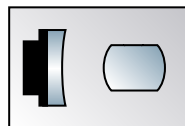
[c-opaff.svg](#)



Nonlinear cavity flat/curved

[c-opafc.png](#)

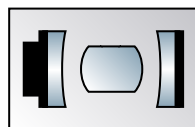
[c-opafc.svg](#)



Nonlinear cavity curved/curved

[c-opacc.png](#)

[c-opacc.svg](#)



Nonlinear kerr cavity

[c-opakerr.png](#)

[c-opakerr.svg](#)



Flipmirror

[c-flip.png](#)

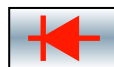
[c-flip.svg](#)



Optical diode green

[c-diodegrn.png](#)

[c-diodegrn.svg](#)



Optical diode red

[c-diodered.png](#)

[c-diodered.svg](#)



3-axes PZT mirror

[c-mirpzt3ax.png](#)

[c-mirpzt3ax.svg](#)