

# Icon Entwicklung

## Unterlagen für den Workshop

### Designing Interactions in XS

Stand März 2012

Carolin Baier  
Stefan Bauerschäfer  
Anja Guse  
Beate Mittelmeyer  
Robert Richter  
**Alexander Schierhorn**  
Clara Weyhenmeyer  
Prof. Dr. Christine Goutrié

Nutzungsumfeld und -situationen  
Kategorisierung von Geräten  
Eingabe für mobile Geräte  
Displaytechnologien  
Auflösung von Displays und Pixelbildern  
Graphical User Interfaces  
Navigationsstrukturen  
Mentale Modelle und Metaphern  
**Icon Entwicklung**  
Grafische Gestaltung: Farben  
Grafische Gestaltung: Schriften

**T H I N K C R O S S - C H A N G E M E D I A**

# Icon Entwicklung

## Definition

Die englische Bezeichnung *Icon* kommt aus dem Griechischen und bedeutet soviel wie Bild. Im Computerbereich wird Icon vielfach mit *Piktogramm* übersetzt.

Ein *Piktogramm* (lat. *pictum* = Bild; griech. *graphein* = schreiben) ist ein einfaches und eindeutiges Bildzeichen, das eine verbale Bezeichnung ersetzt.

Ein Icon soll eine klar definierte Information in sich selbst tragen und sie direkt jedem Betrachter eindeutig übermitteln können. Erst im Kontext der Nutzungsumgebung entfaltet das Icon seine Möglichkeiten. Durch Icons können Funktionen und Informationen abstrahiert und visualisiert werden.

In der Semiotik, der Wissenschaft von Zeichen und ihren Bedeutungen, wird das Icon als ein Zeichen definiert, das dem Darzustellenden ähnelt, und das Symbol als ein Zeichen, das ohne direkten Bezug zum Dargestellten ist.

Nach dem Semiotiker Charles Peirce ist ein *Symbol* hingegen ein willkürlich festgelegtes Zeichen, das nicht durch dessen Abbildung schlüssig ist, sondern durch eine Regel festgelegt wird.

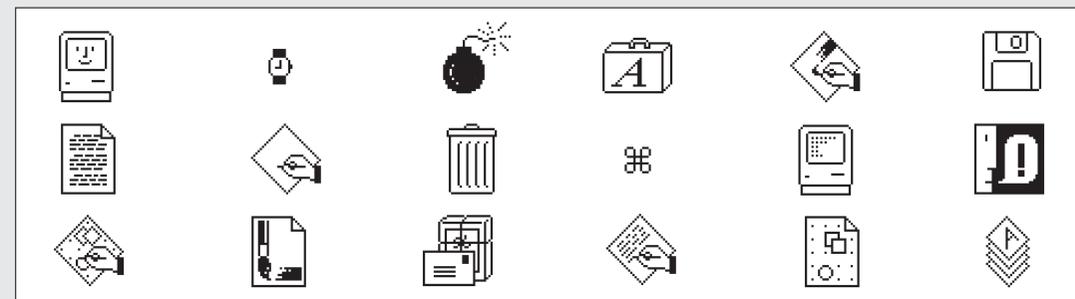
Als Zeichen-Beispiel ohne direkten Bezug zum Dargestellten gelten die Tastensymbole zur Bedienung eines DVD-Players, übernommen von einem analogen Bandlaufwerk. Eine DVD wird jedoch nicht mehr *vorgespuilt*. Für folgende Nutzergenerationen wird die Bedeutung derlei Symbolik nicht mehr durch die Funktionalität des Gerätes nachvollziehbar sein.



Ein Beispiel für ein Piktogramm als Fluchtwegbeschilderung  
(Quelle: <http://upload.wikimedia.org/wikipedia/commons/4/43/Rettungsweg-Links.svg>, Rev. 2012-02-26)



Ein Beispiel für ein Symbol im Straßenverkehr  
(Beginn einer Zone zur Verkehrsberuhigung durch Geschwindigkeitsbegrenzung auf 30 km/h).



Icons von Susan Kare - entwickelt für den Apple Macintosh (Quelle: <http://kare.com>, Rev. 2012-02-26)

# Icon Entwicklung

## Größe und Auflösung

Icons sind meist quadratisch und haben relative Standardgrößen. Die Kantenlängen betragen üblicherweise 16, 24, 32, 48, 64 oder 128 px. Die jeweilige Verwendung ist geräteabhängig.

Es besteht ein direkter Bezug zur Auflösung des Displays auf dem die Icons angezeigt werden sollen. Als einfache Faustregel gilt, je höher die Auflösung, desto größer sind auch die Pixelmaße der Icons.

Ein Beispiel verdeutlicht das:

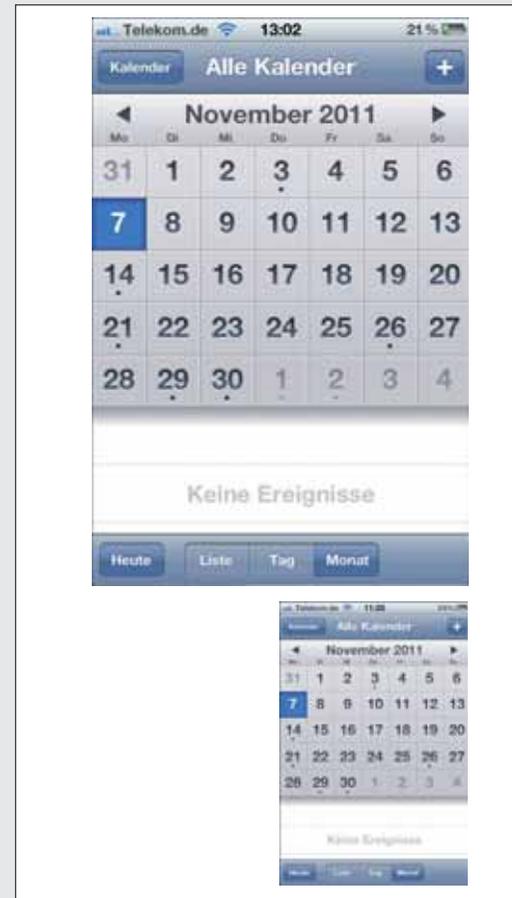
Man nehme an, man entwickelt einen Screen mit diversen Icons für Apples iOS. Die Bildschirme der iPhones 1 bis 3 und 4/4s haben verschiedene Auflösungen. Das iPhone3 hat eine Größe von 320 x 480 px bei 163 ppi und das iPhone4 hat eine weitaus höhere mit 640 x 960 px bei 326 ppi. Beide Telefontypen haben aber dennoch die gleiche physikalische Displaygröße von diagonal 3,5 inch. Man blickt bei beiden Geräten auf die gleichen Icongrößen, obwohl diese beim iPhone3 57 px Seitenlänge haben und beim iPhone4 114 px.

Entwickelt man Icons, so ist es sehr wichtig, die Auflösung des Endgerätes zu kennen.

Arbeitet man z.B. an einem Screen für das iPhone4, so sind die Icons natürlich wesentlich größer als die für ältere iPhone3 mit geringerer Auflösung des Displays. Die Dimensionen der beiden Beispielscreens sieht man in der rechten Abbildung im Vergleich zur tatsächlichen physikalischen Größe auf dem Endgerät selbst.

Man gestaltet zwar für die gleiche Displaygröße, jedoch in gänzlich unterschiedlichen Pixelgrößen.

Zur Berechnung der Auflösung siehe Kapitel 'Auflösung von Displays'.



Größenverhältnisse: iPhone4 Screen (640 x 960 px) und darunter iPhone3 Screen (320 x 480 px).



Zum Größenvergleich ein Ausschnittfoto von einem Flachbildschirm und einem davor gestellten iPhone. Durch unterschiedliche Bildschirmauflösungen kommen derlei Größenunterschiede zustande trotz gleicher Pixelzahlen.

# Icon Entwicklung

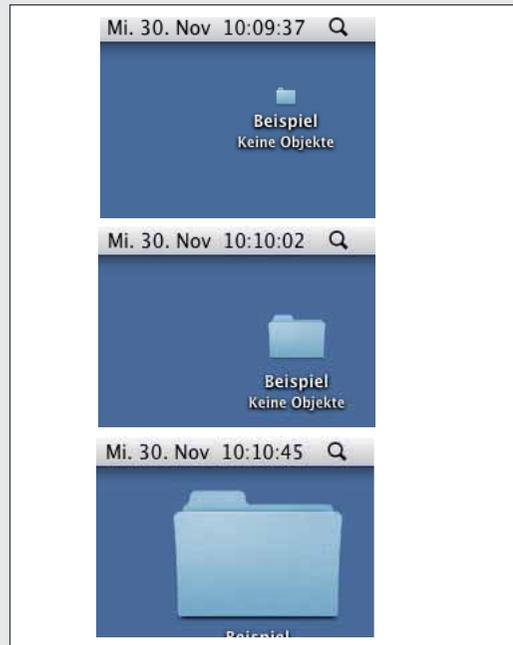
## Eigenschaften

### Skalierbarkeit

Auf mobilen Geräten ist es im Vergleich zu Desktopbetriebssystemen wie Apple MacOS oder Microsoft Windows nicht üblich, skalierbare Icons im ICO-Format zu verwenden.

### Farben

Beschränkungen in der Menge des verwendbaren Arbeitsspeichers ließen früher nur eine reduzierte Farbanzahl zu - der heutige Standard bewegt sich größtenteils bis zu 16,7 Mio. Farben durch weitaus leistungsfähigere Displaytechnologien.



Beispiel eines frei skalierbaren Icons auf dem MacOS Schreibtisch

### Dateiformate

Verschiedene Formate finden Verwendung bei Icons.

- **SVG (Vektorgrafik)**

*Das Format Scalable Vector Graphics ist für zweidimensionale Vektorgrafiken. SVG ist ein XML-basiertes Dateiformat, dessen Dateien können in Texteditoren bearbeitet werden. Durch die Vektoren als grafische Elemente gibt es keine Auflösung im eigentlichen Sinne.*

- **ICO (Pixelgrafik)**

*ICO ist das Format, das unter MS Windows zur Speicherung der Icon verwendet wird. ICO-Dateien können mehrere Einzelbilder enthalten, die das System je nach benötigter Auflösung und Farbtiefe verwendet. Linux oder MacOSX können auch damit umgehen. Windows XP verwendet Icon mit 48 × 48 px. Aktuelle Windows-Versionen unterstützen eine Größe von bis zu 256 × 256 px und 16,7 Mio. Farben.*

- **PNG**

*Das Format Portable Network Graphics ist ein Grafikformat für Rastergrafiken mit verlustfreier Bildkompression. PNG unterstützt neben unterschiedlichen Farbtiefen auch Transparenz per Alphakanal.*

### Anmerkung

Welche Dateiformate benutzt werden, hängt letztlich davon ab, auf welchem Betriebssystem und Endgerät die Anwendung läuft. Favicons werden im .ico-Format erstellt - dazu später mehr.

Darüber hinaus ist auf die Qualität zu achten. Skalierbare Icons bringen da den immensen Vorteil des verlustfreien Skalierens. PNG-Dateien sind durch bessere Algorithmenstrukturen optimaler komprimiert als z.B. das bekannte JPG-Format. Die Anzeige des PNG-Formates in aktueller Software ist inzwischen auch durchweg gegeben.

# Icon Entwicklung

## Funktion und Bedeutung

Icons haben immer eine Bedeutung, also eine Funktion. Nutzer müssen diese in den meisten Fällen erst erlernen. Durch Zuhilfenahme Mentaler Modelle können diese Prozesse vereinfacht werden.

Es gibt jedoch auch intuitive Icons, deren Bedeutung sich wesentlich schneller erschließt durch ein nichtabstraktes Funktionskonzept.

### Zwei Beispiele

In der linken Abbildung sind zwei Icons vom Startscreens des iPhone4 zu sehen.

Zum einfachen Erlernen der *Telefonfunktion* sollte die Funktion möglichst intuitiv zu erkennen sein. Dieses Erkennen geschieht durch mentale Modelle und deren Anwendung durch den Gerätenutzer.

Mentale Modelle können unterschiedlich sein. Ein Palm-Nutzer verbindet mit dem Prozess des Telefonierens ein anderes mentales Modell als ein iPhone-Nutzer. Die Älteren unter uns erinnern sich ja noch an

die Wählscheibe zur Nummereingabe. Mentale Modelle sind einer steten Veränderung unterzogen.

Wenn Nutzer bereits Assoziationen zu einer *Sprechblase* haben, erschließt sich der Kontext leichter. Sie greifen dabei auf erlernte Metaphern zurück. In diesem Fall ist es die Sprechblasen-Metapher, die gesprochene Worte versinnbildlicht und u.a. in Comics adaptiert ist.

Als ein weiteres Beispiel sind in der rechten Abbildung drei Icons aus der Symbolleiste von Microsoft Word 2003 zu sehen. Icons in Symbolleisten von Desktopprogrammen haben meist die Bedeutung einer Schaltfläche.

Die Versinnbildlichung von Programmfunktion oder dazugehöriger Befehle ist die Funktion des Icon.



Intuitiv ist hier links die Telefonfunktion hinterlegt durch Abbildung des Telefonhörers, obwohl kein Hörer mehr „abgenommen“ werden muss bei Mobiltelefonen.

Das rechte Icon zeigt eine Sprechblase und erst der Untertitel erschließt die Bedeutung - hier werden alle Kurznachrichten angezeigt.



Beispiele für weitere intuitive Programmfunktionen:

- Neues, leeres Dokument anlegen in Form eines leeren und unbeschriebenen Blattes
- Öffnen eines vorhandenen Dokumentes durch Visualisierung einer geöffneten Ablagemappe
- farbliches Markieren eines ausgewählten Textabschnittes im Stil eines Textmarkers

# Icon Entwicklung

## Verwendung/Interaktion

Icons die eine programminterne Funktion auslösen, benötigen auf Desktopsystemen nur einen einfachen Klick durch ein Tippen auf dem Touchpad oder mit der Maus. Wird jedoch eine Funktion auf dem Schreibtisch oder Desktop genutzt, wie z.B. das Öffnen eines Ordners, so ist ein Doppelklick notwendig.

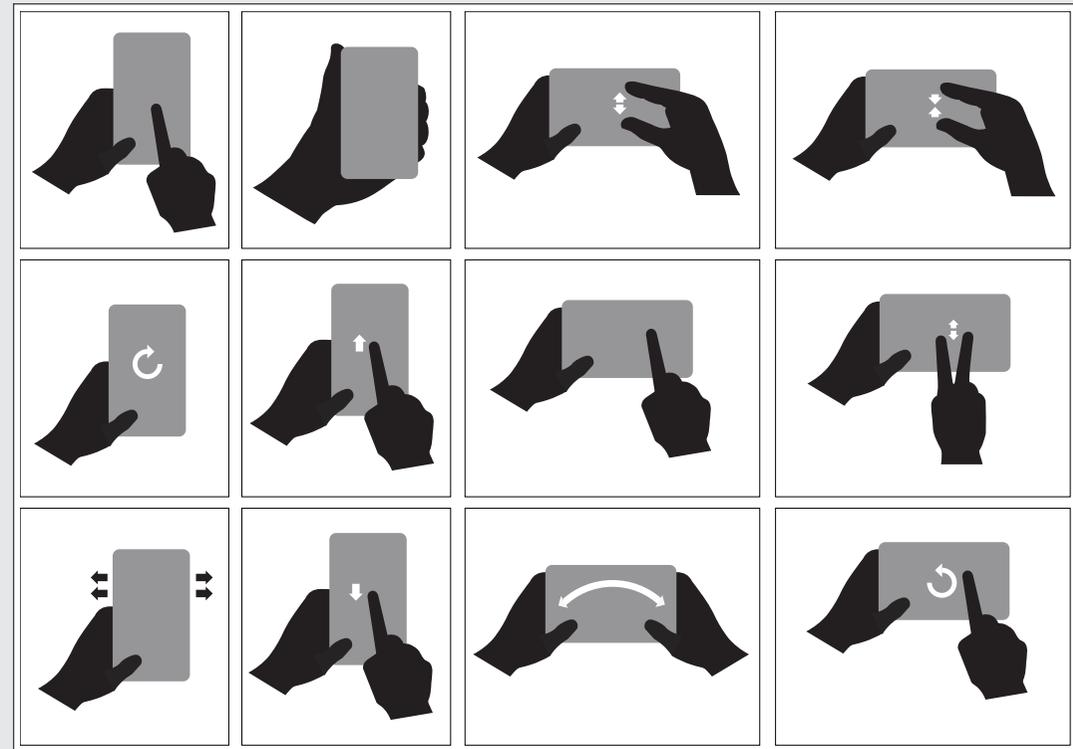
Hover-Effekte gibt es nicht bei Smartphones. Ein Hover-Effekt ist das Verändern eines Links in der Farbe wenn man auf ihn zeigt. Wenn der Link eine Grafik oder ein Icon ist, kann dieses ebenfalls geändert werden. Es geht dabei um visuelles Feedback.

Ebenso gibt es bei mobilen Endgeräten keine Doppelklick-Möglichkeit. Bei dieser Gruppe von Geräten wird per Tasten oder dem berührungsempfindlichem Bildschirm bedient.

Bei der Tastenbedienung von mobilen Endgeräten wird das Icon ausgewählt und dann die Funktion ausgelöst durch eine separate Bestätigungstaste.

Wird das Gerät über ein berührungsempfindliches Display bedient, so ist ebenfalls keine Doppelklickfunktion möglich. Der Grund dafür ist im Bedienkonzept begründet. Es basiert auf einer praktisch zeigerlosen Bedienung mit einer beschränkten Aktionsauslösung durch einen *Klick*.

Eine erweiterte Tastenbedienung ist bei sehr vielen mobilen Geräten möglich durch eine Doppelbelegung der Tasten. Tasten sind in beschränktem Umfang vorhanden und somit prädestiniert für eine Mehrfachbelegung. Das geschieht vielfach durch *längeres Drücken*. Beispielsweise führt auf iPhone-Geräten der letzten zwei Modellreihen ein langer Druck auf die Home Taste zur Sprachsteuerung.



Beispiele von Interaktionen an mobilen Endgeräten.  
(Quelle: <http://gesturecons.com>, Rev. 2012-02-26)

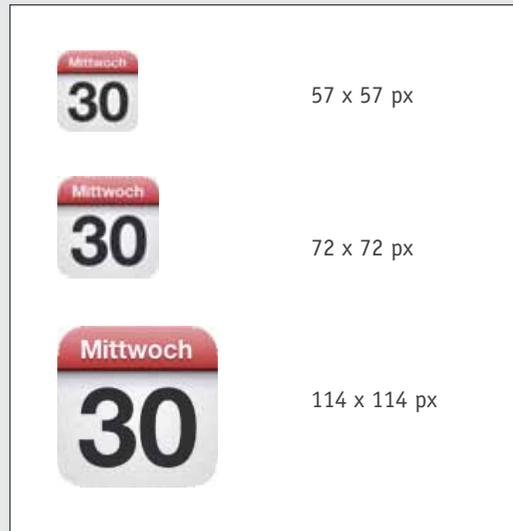
# Icon Entwicklung

## Erstellung

Abhängig vom Endgerät ist die Pixelgröße festzulegen, in der die Icons entwickelt werden.

Bei der Erstellung für beispielsweise iOS Geräte gilt aufgrund der verwendeten Auflösung folgende Größenübersicht:

- iPhone und iPod Touch  
(320 x 480 px bei 163 ppi)  
57 x 57 px - App-Icon  
29 x 29 px - Icon für Einstellungen zur App und Suchergebnisse
- iPad und iPad2  
(1024 x 768 px bei 132 ppi)  
72 x 72 px - App-Icon
- iPhone 4  
(640 x 960 px bei 326 ppi)  
114 x 114 px - App-Icon  
58 x 58 px - Icon für Einstellungen zur App und Suchergebnisse



Beispiel für ein App-Icon anhand der Kalenderfunktion vom Apple iOS:

- iPhone und iPod Touch
- iPad und iPad2
- iPhone 4

Für Android-Geräte sind wiederum andere Größen notwendig für das App-Icon bei unterschiedlichen Bildschirmauflösungen:

- 36 x 36 px für 120 dpi
- 48 x 48 px für 160 dpi
- 72 x 72 px für 240 dpi
- 96 x 96 px für 320 dpi

Mehr zu Android: [http://developer.android.com/guide/practices/ui\\_guidelines/icon\\_design.html](http://developer.android.com/guide/practices/ui_guidelines/icon_design.html)

Der Entwickler von Icons muss sich an den am Markt verbreitesten Geräten orientieren. Vielfach kann man in Web-Apps oder Websites zoomen. Sofern keine Vektor-Icon verwendet werden konnte, ist es empfehlenswert die Icons größer anzulegen um keine Qualitätsverluste zu sehen.

Sind Icons erstellt worden, kann zum Testen der Funktionserkennbarkeit durch das Aussehen des Icon eine Nutzerbefragung sinnvoll sein. Die Zuordnungshäufigkeit zwischen Icon und Funktion bzw. deren Assoziation zum Icon ist dabei relevant.

### Software

Icons können mit speziellen Grafikeditoren, als auch in aktuellen Bildbearbeitungsprogrammen erstellt und bearbeitet werden.

Grafikanwendung für MacOS und Win:

- Adobe Photoshop (<http://www.adobe.com/de/products/photoshopfamily.html>)
- Gimp ([www.gimp.org](http://www.gimp.org)) ist auch als Linux-Version verfügbar

Icon-Anwendungen für Windows:

- Articon ([www.aha-soft.com/articons](http://www.aha-soft.com/articons))
- Pixelformer ([www.qualibyte.com/pixelformer](http://www.qualibyte.com/pixelformer))

Icon-Anwendungen für MacOS:

- Img2icns ([www.img2icnsapp.com](http://www.img2icnsapp.com))

# Icon Entwicklung

## Iconfamilien

Bei der Entwicklung von Icons wird man an den Punkt gelangen, an dem man eine Sammlung, Reihe oder Serie von Icons erstellen möchte.

Diese Zusammenstellung wird *Icon-Familie* genannt.

Grund für die Erstellung einer solchen Familie kann neben der inhaltlichen Zusammengehörigkeit die bessere Anpassung an konzipierte User Interfaces sein oder auch die besondere Anforderung an die Selbsterklärungsfähigkeit der Icons um möglichst auf Texte zur Erklärung verzichten zu können.

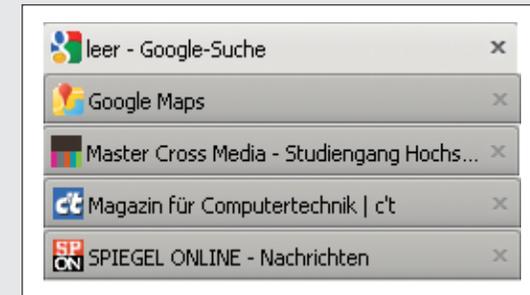
Eine Icon-Familie hat viele Gemeinsamkeiten:

- gleiches Grundraster - gleiche Aufteilung und Ausrichtung der Inhalte
- gleiche Abmaße und Dimension
- einheitlicher Iconstil
- gleiches Darstellungsschema für maximale Aufmerksamkeit auf die Funktionen - dazu gehört u.a. die Strichstärke, die Flächenfüllung, das Farbschema, die Schattierung



Icon-Familie erstellt zu den Olympischen Spielen 2008 in Peking von Estefan Cuanalo „icaro“

(Quelle: <http://icarograf.deviantart.com/art/Olympic-Icons-95189080>, Rev. 2012-02-26)



Beispiel für Favicons in Mozilla Firefox Browsertabs.

### FAVICON

Über die Grenzen von Betriebssystemen hinweg findet das ICO-Format zusätzlich Anwendung beim Favicon.

Das Favicon steht für favorite icon und ist ein kleines, 16×16 oder 32×32 px großes Icon, Symbol oder Logo, das unter anderem in der Adresszeile eines Browsers links von der URL angezeigt wird und meist dazu dient, die zugehörige Website auf wiedererkennbare Weise zu kennzeichnen. Von den Browsern werden aber nur 16×16 px angezeigt. Das Favicon taucht auch in der Lesezeichenleiste (Favoriten), bei den Registerkarten und unter Windows beim Speichern einer Seite auf dem Desktop auf.