

Teil 1: In der Theorie

Design Language

Teil 2: In der Anwendung

Mac vs. Anti-Mac

media
computing
group



Proseminar: Human Computer Interaction

Design Language

Negah Nabbi
Patrick Zimmer

Teil 1: Designsprache in der Theorie

- Einleitung und Information über die Autoren
- Designsprache im Vergleich zum klassischen Verständnis der Sprache
- Alltäglicher Gebrauch von Designsprachen
- Designsprache und ihre Bedeutung für Objekte und ihre Benutzer
- Die Struktur einer Designsprache
- Entwicklung einer Designsprache



Information über die Autoren



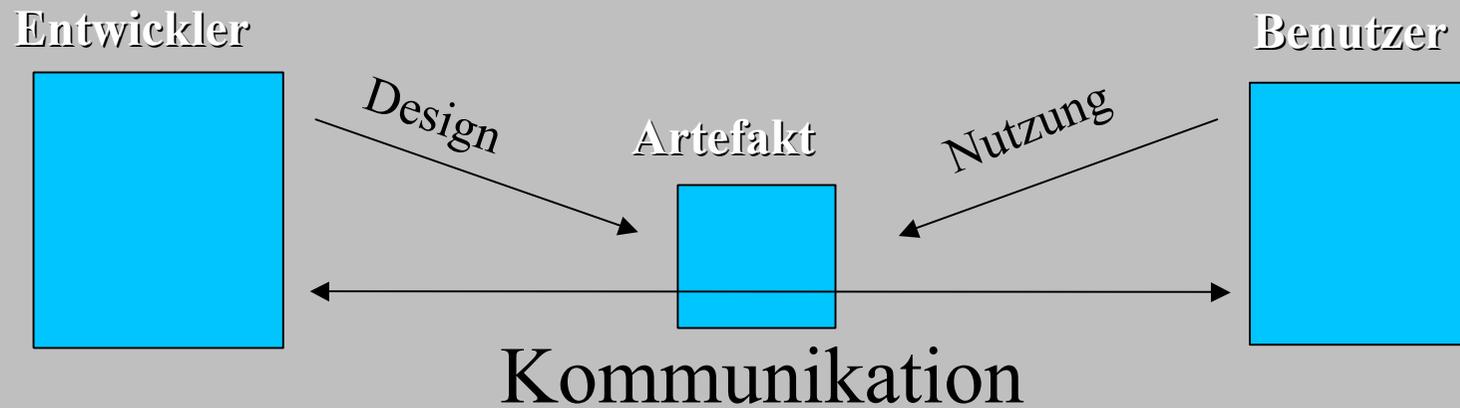
John Rheinfrank
Product Designer



Shelly Evanson
Graphic Designer

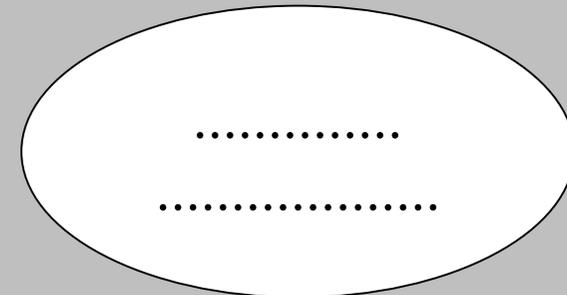


Einleitende Definition der Designsprache



Gemeinsamkeiten zwischen Designsprachen und natürlicher Sprache

- Die Grundelemente: Buchstaben und Wörter bzw. Formen, Vorgänge, ...
- Regeln zur Komposition (Grammatik)
- Vermittlung von mehr als nur der Information
- Intuitives Verständnis und Benutzung
- Hohe Integration in den Alltag
- Ständige Entwicklung (Evolution)



Designsprachen im alltäglichen Gebrauch

Designsprachen sind allgegenwärtig und entwickeln sich meist unterbewusst

Designsprachen werden daher intuitiv und nicht strukturell erlernt

Folgend einige Beispiele...

media
computing
group



Proseminar: Human Computer Interaction

Design Language

Negah Nabbi
Patrick Zimmer

Designsprachen im alltäglichen Gebrauch



C83/6374

media
computing
group



Proseminar: Human Computer Interaction
Design Language

Negah Nabbi
Patrick Zimmer

Designsprachen im alltäglichen Gebrauch



Designsprachen im alltäglichen Gebrauch



Designsprachen im alltäglichen Gebrauch



Designsprachen im alltäglichen Gebrauch



media
computing
group



Proseminar: Human Computer Interaction
Design Language

Negah Nabbi
Patrick Zimmer

Designsprachen im alltäglichen Gebrauch



Artefakte und ihre Bedeutung

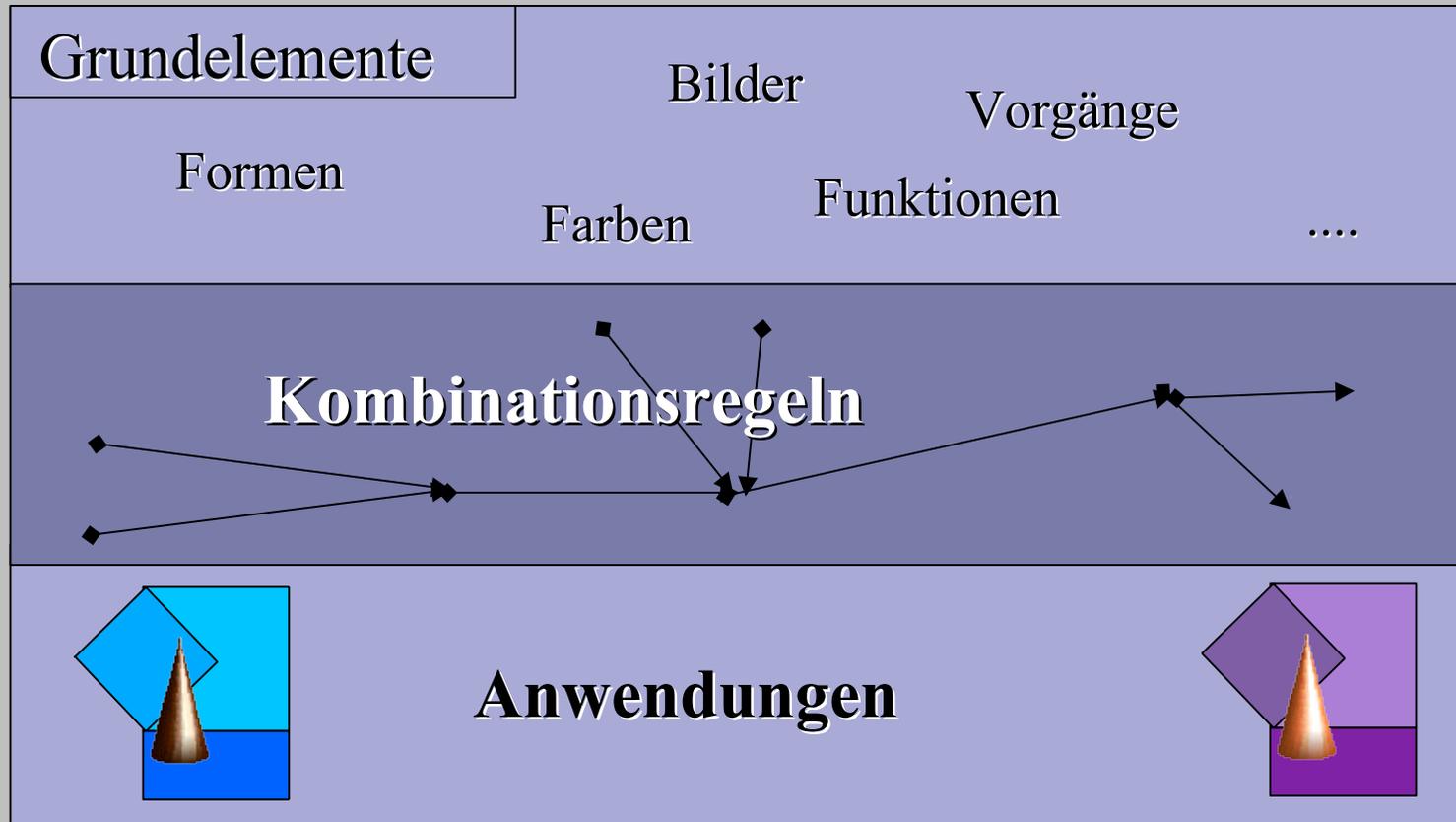
Der Designer schafft Bedeutung

Bedeutung ist **nicht** gleich Nutzen

Der Benutzer leitet sich von der Bedeutung
seinen Nutzen des Objektes ab



Struktur einer Designsprache



Konstruktion einer Designsprache

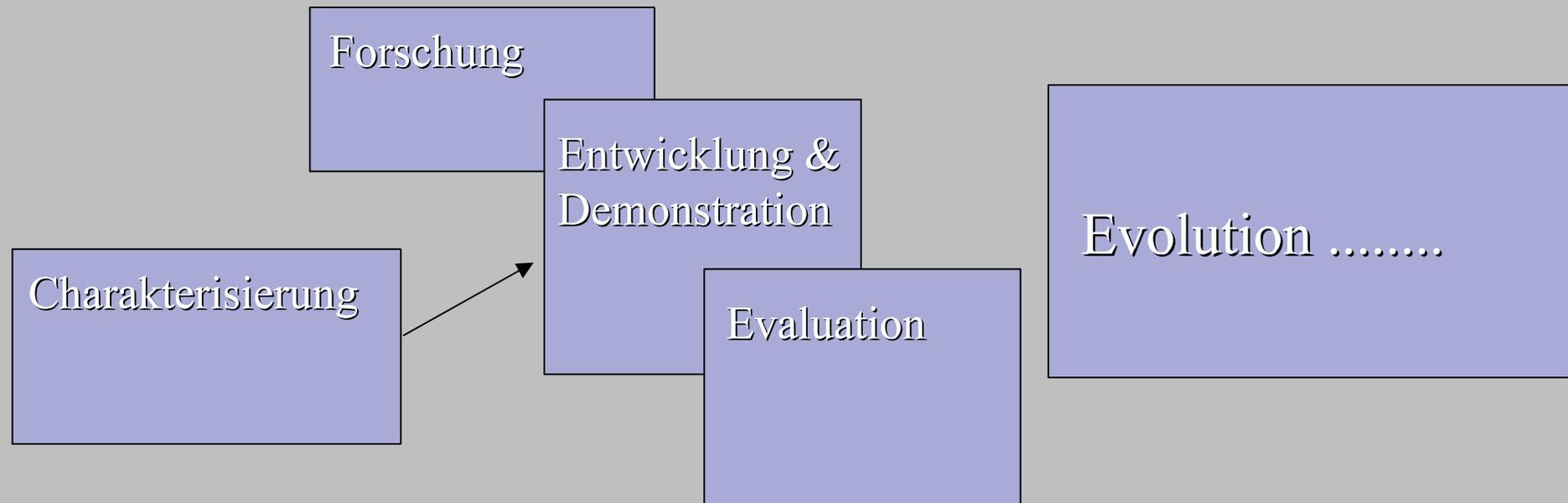
Schritt 1

Schritt 2

Schritt 3

Schritt 4

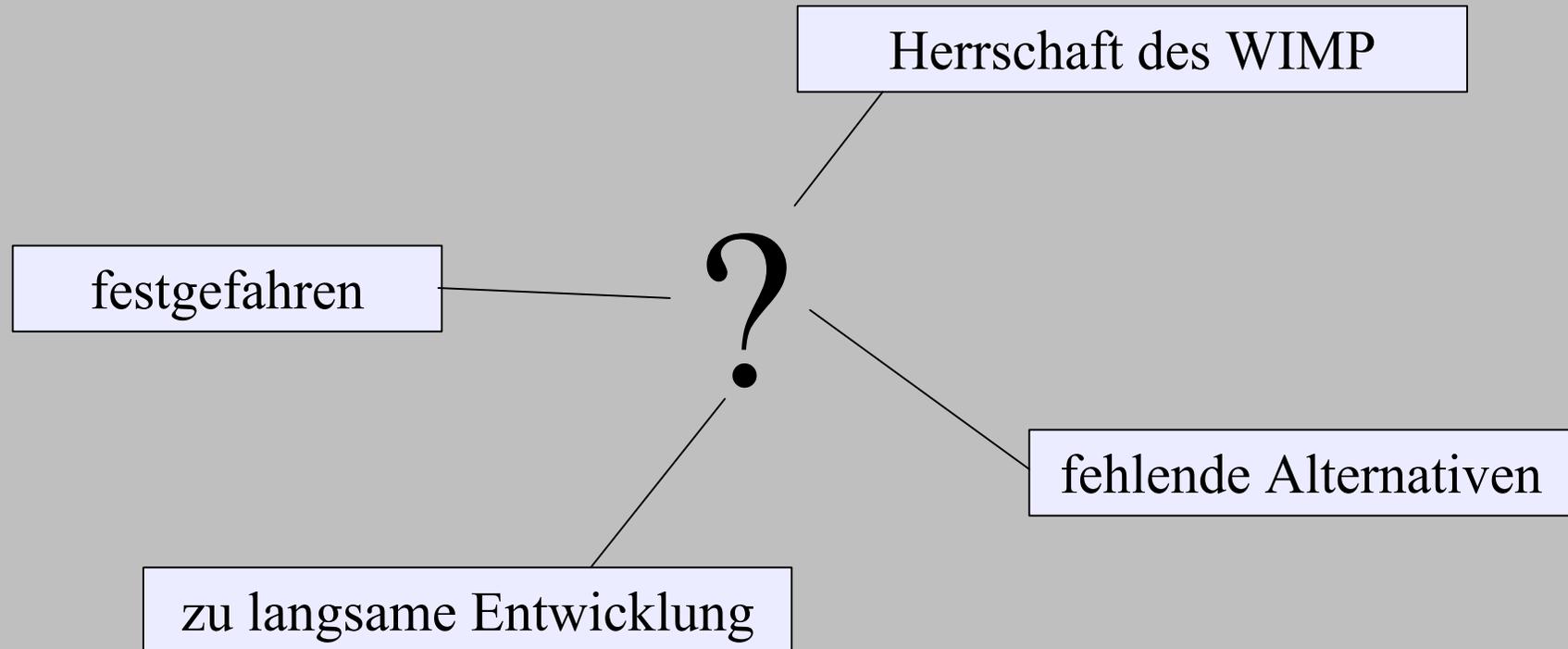
Schritt 5



Probleme der Designsprache bezogen auf heutige Interfaces



Probleme der Designsprache bezogen auf heutige Interfaces



Teil 2: Mac vs. Anti-Mac

- Einführung und Übersicht
- Die Prinzipien des Mac und Gründe, sie zu brechen
- Das neue Interface mit Anti-Mac Prinzipien



Die Macintosh Human Interface Guidelines

(HI-Guide)

- Erstmals veröffentlicht 1987 durch Apple
- Pflichtlektüre für alle Apple Programmierer
- Zweigeteilt; Teil 1: Leitprinzipien; Teil 2: Interfaceelemente
- Teil 2 wird ständig aktualisiert für jedes Mac OS, während Teil 1 seit jeher unangetastet ist.



Die Macintosh Human Interface Guidelines - zum zweiten Teil: Die Interfaceelemente

Ein sehr trockenes Thema, wo doch das Aussehen eines klassischen Interfaces im allgemeinen bekannt ist. Im zweiten Teil des HI-Guides werden alle möglichen Elemente beschrieben und deren Darstellung und Anordnung vorgeschrieben.

Es ist vorgeschrieben, wie Menüs aufgebaut sind, welche Arten von Menüs es gibt, welche Elemente sie enthalten dürfen und welche nicht, wie groß der Zeilenabstand sein darf, welche Schriftarten und Typen verwendet werden dürfen, welche Untermenüs es geben darf.

Die Erscheinungsform, Größe und Position von Schaltflächen ist festgelegt, wie viele davon und genau welche in einem Dialog vorkommen dürfen, wieviele Pixel der Abstand zwischen ihnen beträgt und natürlich was sie bewirken

Das Aussehen von Fenstern, die Platzierung und Form der Buttons und der Scrollleisten, die Rahmenbreite, der Platz für Menüs und sonstiger Elemente ist natürlich auch genau festgelegt.

Das Ganze erstreckt sich über zig Seiten und es gibt wohl nichts, was dort nicht auf genaueste definiert ist....



Die Macintosh Human Interface Guidelines - zum zweiten Teil: Die Interfaceelemente

Ein sehr trockenes Thema, wo doch das Aussehen eines klassischen Interfaces im allgemeinen bekannt ist. Im zweiten Teil des HI-Guides werden alle möglichen Elemente beschrieben und deren Darstellung und Anordnung vorgeschrieben.

Es ist vorgeschrieben, wie Menüs auszusehen sind, welche Arten von Menüs es gibt, welche Elemente sie enthalten dürfen und welche nicht, wie groß der Zeilenabstand sein darf, welche Schriftarten und Typen verwendet werden dürfen, welche Unterpunkte geben darf.

Die Erscheinungsform, Größe und Position von Schaltern ist festgelegt, wie viele davon und genau welche in einem Dialog vorkommen dürfen, wieviele Pixel der Abstand zwischen ihnen beträgt und natürlich was sie bewirken

Das Aussehen von Fenstern, die Platzierung und Form der Buttons und der Scrollleisten, die Rahmenbreite, der Platz für Menüs und sonstiger Elemente ist natürlich auch genau festgelegt.

Das Ganze erstreckt sich über zig Seiten und es gibt wohl nichts, was dort nicht auf genaueste definiert ist....



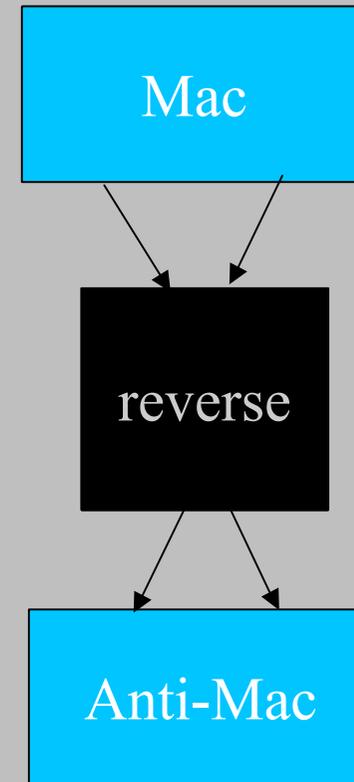
Ersatzprogramm: Der Anti-Mac



Don Gentner



Jakob Nielsen



Umkehrung aller Mac-Leitprinzipien



Macintosh - Leitprinzipien

Nummer 1: Metaphern (Metaphors)

Die Idee: Der Bezug zu reellen Objekten soll das Erkennen und Erlernen der Funktionen erleichtern

Die Gefahr: Funktionalität wird nur unzureichend oder falsch erkannt. Vorfestlegung auf bekannte Objekte schränkt zu sehr ein



Macintosh - Leitprinzipien

Nummer 1: Metaphern (Metaphors)

Die Idee: Der Bezug zu reellen Objekten soll das Erkennen und Erlernen der Funktionen erleichtern

Die Gefahr: Funktionalität wird nur unzureichend oder falsch erkannt. Vorfestlegung auf bekannte Objekte schränkt zu sehr ein



Macintosh - Leitprinzipien

Nummer 2: Direktzugriff (Direct Manipulation)

Die Idee: Atomarer Zugriff auf einzelne Objekte

Die Gefahr: Fehlende Funktionalität für die
Behandlung von Objektgruppen unter komplexen
Bedingungen



Macintosh - Leitprinzipien

Nummer 2: Direktzugriff (Direct Manipulation)

Die Idee: Anularer Zugriff auf einzelne Objekte



Die Gefahr: Fehlende Funktionalität für die
Behandlung von Objektgruppen unter komplexen
Bedingungen



Macintosh - Leitprinzipien

Nummer 3: Zeiger (See-And-Point)

Die Idee: Was der Benutzer sieht, kann er anklicken, um Informationen zu erhalten

Die Gefahr: Einschränkung auf sichtbare Objekte.
Verlust der Fähigkeit zur komplexen Kommunikation



Macintosh - Leitprinzipien

Nummer 3: Zeiger (See-And-Point)

Die Idee: Was der Benutzer sieht, kann er anklicken, um Informationen zu erhalten.

Die Gefahr: Einschränkung auf sichtbare Objekte.
Verlust der Fähigkeit zur komplexen Kommunikation



Macintosh - Leitprinzipien

Nummer 4: Konsistenz (Consistency)

Die Idee: Leichtere Wiedererkennung von Strukturen zum schnelleren Lernen

Die Gefahr: Verlust der Informationsvielfalt. Keine Registrierung von Details



Macintosh - Leitprinzipien

Nummer 4: Konsistenz (Consistency)

Die Idee: Leichtere Wiedererkennung von Strukturen zum schnelleren Lernen

Die Gefahr: Verlust der Informationsvielfalt, keine Registrierung von Details



Macintosh - Leitprinzipien

Nummer 5: „WYSIWYG“ (What you see is what you get)

Die Idee: Live-Manipulation. Direkt präsentierte Ergebnisse

Die Gefahr: Das „Warum?“ geht bei den meisten Informationen verloren. Schlecht für Nachbearbeitungen oder Projekte mit mehreren Beteiligten



Macintosh - Leitprinzipien

Nummer 5: „WYSIWYG“ (What you see is what you get)

Die Idee: Live Manipulation. Direkt präsentierte Ergebnisse

Die Gefahr: Das „Warum?“ geht bei den meisten Informationen verloren. Schlecht für Nachbearbeitungen oder Projekte mit mehreren Beteiligten



Macintosh - Leitprinzipien

Nummer 6: Benutzerkontrolle (User Control)

Die Idee: Der Benutzer hat volle Kontrolle über sein System

Die Gefahr: Viele Aufgaben sind zu zeitaufwendig oder zu komplex



Macintosh - Leitprinzipien

Nummer 6: Benutzerkontrolle (User Control)

Die Idee: Der Benutzer hat volle Kontrolle über sein System

Die Gefahr: Viele Aufgaben sind zu zeitaufwendig oder zu komplex



Macintosh - Leitprinzipien

Nummer 7: Informativität (Feedback and Dialog)

Die Idee: Der Benutzer wird über alles informiert.
Dies geht einher mit der Benutzerkontrolle

Die Gefahr: Flut von irrelevanten Informationen



Macintosh - Leitprinzipien

Nummer 7: Informativität (Feedback and Dialog)

Die Idee: Der Benutzer wird über alles informiert.
Dies geht einher mit der Benutzerkontrolle

Die Gefahr: Flut von irrelevanten Informationen



Macintosh - Leitprinzipien

Nummer 8: Absicherung (Forgiveness)

Die Idee: Alle Vorgänge lassen sich rückgängig machen; vor unwiederrufbaren Aktionen wird gewarnt

Die Gefahr: Oft wird der Benutzer wiederholt durch diverse Dialoge beeinträchtigt



Macintosh - Leitprinzipien

Nummer 8: Absicherung (Forgiveness)

Die Idee: Alle Vorgänge lassen sich rückgängig machen, vor unwiederrubaren Aktionen wird gewarnt

Die Gefahr: oft wird der Benutzer wiederholt durch diverse Dialoge beeinträchtigt



Macintosh - Leitprinzipien

Nummer 9: Stabile Umgebung (Perceived Stability)

Die Idee: In der Abwesenheit des Benutzers finden keine Veränderungen statt

Die Gefahr: Bei vernetzten Systemen unmöglich einzuhalten und unproduktiv

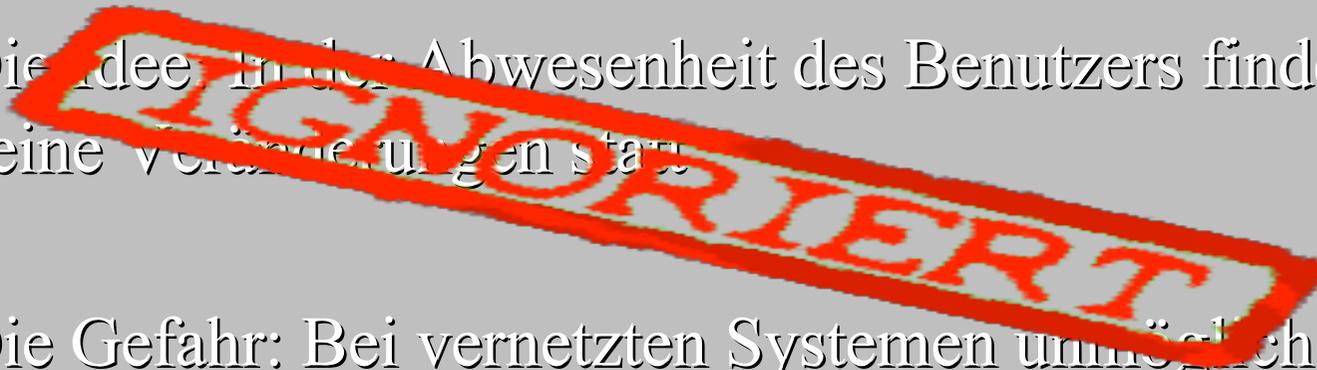


Macintosh - Leitprinzipien

Nummer 9: Stabile Umgebung (Perceived Stability)

Die Idee: In der Abwesenheit des Benutzers finden keine Veränderungen statt.

Die Gefahr: Bei vernetzten Systemen unmöglich einzuhalten und unproduktiv



Macintosh - Leitprinzipien

Nummer 10: Einheitsdesign (Aesthetic Integrity)

Die Idee: Konsistentes Design erleichtert die Navigation

Die Gefahr: Zu eintönige Strukturen wirken uninteressant und verlieren ihre Übersicht bei zu vielen Daten



Macintosh - Leitprinzipien

Nummer 10: Einheitsdesign (Aesthetic Integrity)

Die Idee: Konsistentes Design erleichtert die Navigation

Die Gefahr: Zu eintönige Strukturen wirken uninteressant und verlieren ihre Übersicht bei zu vielen Daten



Macintosh - Leitprinzipien

Nummer 11: Zustandslosigkeit (Modelessness)

Die Idee: Der Benutzer kann von überall alle Aufgaben durchführen

Die Gefahr: Ein Interface ohne Zustandseinteilung ist nicht denkbar. Jedes Programm zum Beispiel ist schon ein Zustand und soll nicht alles können



Macintosh - Leitprinzipien

Nummer 11: Zustandslosigkeit (Modelessness)

Die Idee: Der Benutzer kann von überall alle Aufgaben durchführen

Die Gefahr: Ein Interface ohne Zustandsentwöhnung ist nicht denkbar. Jedes Programm zum Beispiel ist schon ein Zustand und soll nicht alles können



Das Ergebnis: Anti-Mac Leitprinzipien

- Eine zentrale Rolle der Sprache
- Detailliertere Repräsentation von Objekten
- Ein vielfältigeres Interface
- Höherer Anspruch an den Benutzer
- Verteilte Kontrolle



Schlusswort: Zukünftige Interfaces

Es ist höchste Zeit, mit der Entwicklung neuer leistungsfähiger Interfaces zu beginnen um zumindest eine Alternative zum klassischen WIMP Standard zu bieten.



Vielen Dank für eure
Aufmerksamkeit

media
computing
group



Proseminar: Human Computer Interaction
Design Language

Negah Nabbi
Patrick Zimmer